FSI Viewer 5.5

Let your website visitors have a closer look !

User Manual

FSI Viewer • Pages Add-on • Showcase Add-on • Plug-ins

FSI Viewer®
FSI Plug-ins
FSI Pages Add-on
FSI Showcase Add-on

Developed by:

NeptuneLabs GmbH
Detmolder Str. 210a
D-32791 Lage
Germany

Last updated: July 2009

FSI Viewer Version	5.5
FSI Pages	5.5
FSI Showcase Version	5.5
Manual Revision	044

Table of Contents

NeptuneLabs FSI Viewer®

Introduction

FSI Viewer (‘**F**lash based **S**ingle Source **I**mage Viewer’) and the Add-ons FSI Pages and FSI Showcase have been designed to display high resolution images on the internet requiring low bandwidth only. Using the Adobe Flash™ browser plug-in FSI Viewer requests image data from Single Source Imaging Servers.

Using FSI Viewer you can present two-dimensional images as well as three-dimensional, photorealistic presentations consisting of multiple individual images. By using Single Source Imaging Servers only one high resolution source image is required for each image to be displayed.

Based on Adobe Flash™ plug-in FSI Viewer can be integrated almost independently of the browser version. As today the penetration of this plug-in exceeds 97% of all internet users FSI products can be used without downloading or setting up a browser plug-in in almost all cases.

By means of a large number of configuration options and optional FSI Skins, FSI Viewer, FSI Pages Add-on and FSI Showcase Add-on can widely be adjusted to integrate seamlessly into your website. The FSI Plug-in system is capable of integrating additional functionality into FSI Viewer at runtime without increasing the download size in general.

Thank you for using NeptuneLabs software!

www.fsi-viewer.com - Online Resources for FSI Viewer
Visit www.fsi-viewer.com for software updates, regularly updated samples, tutorials and downloads.

Compatibility and User System Requirements

Imaging Servers
FSI products covered by this manual can at present securely be run with FSI Server from version 1.0, eRez Imaging Server from version 2.54, iSeeMedia Zoom Image Server from version 4 and TrueSpectra version 4.1.1.

Adobe Flash™
FSI Viewer, FSI Pages Add-on and FSI Showcase Add-on require Flash™ Plug-in version 6 or newer to be displayed in browsers.

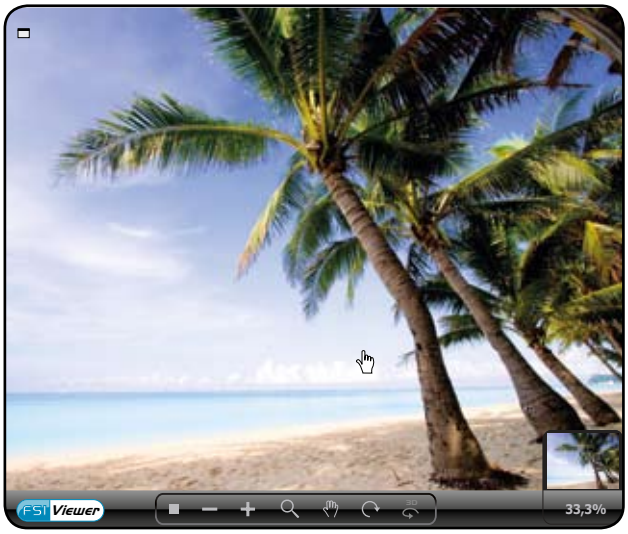
FSI Viewer has been successfully tested with the following software versions:

Adobe Flash™ Plug-in	-	6.0.65.0 to 10.0.22.87
FSI Server	-	1.0
YaWah eRez	-	Imaging Server 2.5.4 – 5.6.16
iSeeMedia	-	ZoomImageServer 4.0 – 4.6
TrueSpectra	-	Image Server 4.1.1 Service Patch 3

Using FSI Viewer

FSI Viewer is navigated by the menu bar (here below the image) and by using the mouse directly on the image. For example, you can select a section of the image you want to magnify.
The mouse functions are determined by the corresponding buttons on the menu bar (zoom, pan, rotate, etc.). The example to the left shows the mouse mode "zoom".

The optional small navigator window (bottom-right) displays the position of the image section currently viewed.



The Menu Bar

The menu bar might look different depending on skins or additional plug-ins.

Main Functions

	**Back to Initial View.** Undoes zoom, pan and rotation and restores the initial view. (identical with pressing space bar)
	Zoom In
	Zoom Out

Mouse Modes

The "Mousemodes" or "MousemodeSelect" plug-in is required to display the following mouse mode buttons.

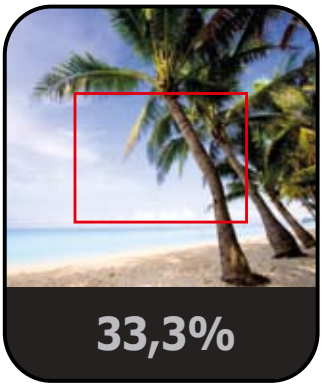
	**Mouse - Zoom In** Choose this function to enlarge segments using the mouse. Click on the image and drag the frame over the desired segment. Alternatively you can click on the image, without marking a segment. The image will then be magnified in steps. To zoom out in steps, hold down the CTRL-key and click on the image.
	**Mouse - Pan** Choose this function to pan the image using the mouse. Click on the image and drag in the desired direction. To return to the original view, hold down the CTRL-key and click on the image.
	**Mouse - Turn** Choose this function to rotate the image around the z-axis. Click on the image and drag in whatever direction you want to turn the object (to the right or the left). To reset the rotation, hold down the CTRL-key and click on the image.
	**Mouse - Rotate (only for 3D presentations)** Choose this function to rotate the object around the y-axis or the x-axis if available. Click on the image and drag to the left or to the right. Move the mouse up or down to rotate the object around the x-axis. To reset rotation, hold down the CTRL-key and click on the image.

Additional Buttons

	**Hide/Display menu** Displays or hides the user interface.
	**Information** Displays information about the viewer and refers to a configurable help page (*Parameter HelpURL*).

Using the navigator window

In the navigator window you can see a miniature presentation of the entire image. The segment which is currently viewed is framed in red. In the navigator window you can change the current segment either by dragging the frame or by clicking on the desired area of the image.



Keyboard Shortcuts

FSI Viewer, FSI Pages and FSI Showcase can additionally be controlled using the keyboard. The list below shows the key function assignments.

Key	Function
Num 5, spacebar	Back to original view
Num 4, left arrow	Move to the left
Num 6, right arrow	Move to the right
Num 8, up arrow	Move up
Num 2, down arrow	Move down
Num 1	Turn around the z-axis to the left
Num 3	Turn around the z-axis to the right
Num 7	Rotate around the y-axis to the left (3D only)
Num 9	Rotate around the y-axis to the right (3D only)
Num /	Rotate around the x-axis (3D only) or previous view
Num *	Rotate around the x-axis (3D only) or previous view
Num 0	Hide/display user interface
D	If you have activated the debug mode, you can hide or display the debug window using this key.
I	Refresh the image information in the debug window.
Pos 1	FSI Pages: First Page
Page Up	FSI Pages: Previous Page
Page Down	FSI Pages: Next Page
End	FSI Pages: Last Page

NeptuneLabs FSI Viewer®

Setup

Setting up FSI Viewer is as easy as copying all FSI Viewer files to your web server. The only file that will directly be accessed by the web browser is fsi.swf located in the main index. FSI Server ships with a readily installed copy of FSI Viewer. If you are using FSI Server, you can skip this chapter and continue reading the next chapter on "Licencing".

As the URL to fsi.swf file is part of each HTML code presenting FSI Viewer it is recommended to keep the path to the file fsi.swf as short as possible by either:

a) Creating a subdirectory for FSI Viewer in the root directory of you web server.

HTML code addressing FSI Viewer in root directory

```
<PARAM name="Movie" VALUE="/fsi/fsi.swf?cfg=foo">
```

This way the file can easily be addressed relatively from any location of your website.

b) Creating a subdomain to access FSI Viewer absolutely.

HTML code addressing FSI Viewer by subdomain

```
<PARAM name="Movie" VALUE="http://fsi.domain.com/fsi.swf?cfg=foo">
```

Index structure of FSI Viewer

Directory	Content Description
core	Core components
debug	Debugging components *This folder can safely be deleted if debugging is not required any more*
languages	User interface language files
plugins	FSI Plug-ins extending FSI Viewer at runtime
skins	FSI Skins containing the visible parts of the user interface

Please Note:
You need to make sure that your web server properly serves files with the following extensions: *.swf, *.fsi, *.xml, *.nlm, *.plg

Setting up FSI Viewer

Step 1

Copy all FSI Viewer files using an FTP program to the corresponding directory on your web server.
Please note that all files have to be transferred in binary mode.

Step 2

Edit the default configuration file "_default.fsi " located in the main index of your setup directory using a text editor of your choice.
The most important modifications are:

- FPXServerType
  Enter the type of your imaging server, e.g. "fsi" or"erez"
- FPXBase
  Request base of the imaging server, e.g.

```
"http://domain/fsi/server?source="
or
"http://zis.domain.com/fif="
```

Additionally you can create configuration files for individual images or use HTTP queries to access different images. Please refer to the chapter "Automated Implementation of Images" if you plan to use a large number of images with FSI Viewer.

Step 3

Add the HTML source for a Flash movie to your website and enter the path to the file "fsi.swf" with the desired parameters as a HTTP query for the parameter "Movie" (object tag) and "Src" (embed tag).

<PARAM NAME="movie" VALUE="fsi.swf?cfg=foo&debug=1">

Please use the files in the "samples" folder of the installation archive as a guide line.

Licencing FSI Viewer

You can use FSI Viewer without a licence key for evaluation purposes. In this case the phrase "Evaluation Copy" appears on top of the image and the about box displays "Unregistered Evaluation Copy" in the "Licenced to" section.

After purchasing an FSI Viewer licence you will receive a licence key that removes the "Evaluation Copy" mark and the name of the licensee appears in the "licenced to" section of the about box.

Entering the Licence Key
After receiving your licence key you need to add the key to your _default.fsi configuration file located in the main index of your FSI Viewer setup directory.
Users of eRez imaging server can alternatively enter the FSI licence key in the eRez Administration Console using the "Enter FSI Licence Key" option.

The general way for all systems to enter the key is:

1. Open the file configuration file "_default.fsi" using a text editor
2. Copy the licence key you received
3. Paste the licence key (including the <licence> tag)
4. Save the modified file

Licence key added to the _default.fsi file

```
<fsi_parameter>

      <licence>
            [Licence key goes here]
      </licence>

      ...
</fsi_parameter>
```

Make sure to flush your browser cache as the _default.fsi file might have been cached by your browser.

Please note that FSI Viewer will run in evaluation mode regardless of the licence key if you access FSI Viewer via a different domain or IP address than you have licenced FSI Viewer to. The same applies to unregistered FSI Plug-ins or FSI Skins.In this case please check the FSI Viewer *debug window*, which provides information on the licencing status and contact your dealer if the problem persists.

NeptuneLabs FSI Viewer

Integration into HTML Pages

Required HTML-Source Code

Object Tag

All recent browsers require an <Object> tag to integrate a Flash movie clip. Older browsers use the obsolete <Embed> tag instead.

Sample HTML code to integrate FSI Viewer into HTML pages: (Variable data printed in bold and square brackets):

HTML integrations for modern and old browsers

```
<Object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.
macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,65,0" width="[Width]"
height="[Height]">
<PARAM name="movie" VALUE="[URL and Parameter]" />
<PARAM name="bgcolor" VALUE="[Background color]" />
<PARAM name="menu" VALUE="false" />
<PARAM name="allowscriptaccess" value="always" />
<PARAM name="allowfullscreen" value="true" />
<PARAM name="quality" value="high" />
</Object>
```

Embed Tag

For browsers (e.g. Netscape 4.x) that do not support the <object>-tag, an additional, encapsulated <embed>-tag with identical parameters is required.

HTML integrations for modern and old browsers

```
<Object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.
cab#version=6,0,65,0" width="[Width]" height="[Height]">
<PARAM name="movie" VALUE="[URL and Parameter]" />
<PARAM name="bgcolor" VALUE="[Background color]" />
<PARAM name="menu" VALUE="false" />
<PARAM name="allowscriptaccess" value="always" />
<PARAM name="allowfullscreen" value="true" />
<PARAM name="quality" value="high" />
<EMBED name="application/x-shockwave-flash" PLUGINSPAGE="http://www.macromedia.
com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash" SRC="[URL and
Parameter]" WIDTH="[Width]" HEIGHT="[Height]" BGCOLOR="[Background color]" MENU="false"
ALLOWSCRIPTACCESS="always" ALLOWFULLSCREEN="true" QUALITY="high"></EMBED>
</Object>
```

Variables in the **<object>** and **<embed>** tags

[Width]	Width of the viewer in pixel or percent
[Height]	Height of the viewer in pixel or percent
[Background color]	Background color, hexadecimal (e.g. #FFFFFF for white)
[URL and parameter]	URL to FSI Viewer and parameters provided as HTTP query, e.g. "?cfg=foo"

For further possible parameters or XHTML configuration options, please refer to Flash documentation or renowned HTML references.

NeptuneLabs FSI Viewer

Configuration

Configuration parameters can be provided using XML- configuration files and/or by HTTP queries appended to the URL of the Flash movie clip.
Providing parameters by HTTP query is recommended if you plan to display large numbers of images (*Automated Implementation of Images*).

In the simplest case you only define the relative path to a configuration file within the "src" parameter of the <object>-Tag using the "cfg" parameter:

Defining the CFG parameter inside the object tag

```
<PARAM name="movie" VALUE="fsi.swf?cfg=[relative path to FSI-file]">
```

The following table lists FSI parameter value types, referred to in this documentation:

Type	Example
Number	"90"
String	"ZoomIn"
URL	"http://www.fsi-viewer.com/"
Boolean	"0" / "1" or "true" / "false"
HexColor	"FF0000"

Using XML Configuration Files (*.fsi)

FSI Viewer uses XML configuration files which can easily be edited with any text editor.

The default configuration file "_default.fsi" - which is located in the same folder as the main viewer file "fsi.swf" – is the most important XML configuration file and will always be evaluated first.

To use FSI Viewer with additional configuration files you have to provide the name of the configuration file using the "cfg" parameter attached to the URL of the "fsi.swf" file:

Specifying an XML configuration file for FSI Viewer

```
PARAM name="movie" VALUE="/fsi/fsi.swf?cfg=foo">
```

Configuration files are structured into configuration groups (XML nodes) containing parameter names and values. Only parameters in these groups will be recognized by FSI Viewer. Configuration files do not need to contain all groups though.
Each parameter has to be provided as an individual XML child node of a group where the node name is the parameter name and the parameter value needs to be provided as "value" attribute of the node.

Structure of XML configuration files

```
<fsi_parameter>

	<Viewer>
		<parameter value="" />
	</Viewer>

	<FPX>
		<parameter value="" />
	</FPX>

	<Plugins>
		<parameter value="" />
	</Plugins>

	<Options>
		<parameter value="" />
	</Options>

</fsi_parameter>
```

Simple XML configuration file

```
<fsi_parameter>
	<FPX>
		<Src value="image.fpx" />
	</FPX>
	<Options>
		<Debug value="1" />
	</Options>
</fsi_parameter>
```

Please keep in mind the following rules when using XML configuration files:

- Value attributes must be provided in quotation marks
- Configuration files might be cached by your browser, so that changes in these files require flushing the browser cache. This does not apply when using the *debug mode*.
- Make sure to save configuration files in Unicode format if your configuration file contains language specific diacritics, Japanese characters and alike

You can comment on or disable sections of an XML configuration file using the usual XML comment syntax:

Comments and disabling parameters temporarily

```
<FPX>
	<!-- This is a comment -->
	<Src value="image.fpx" />

	<!-- The following section will be ignored -->
	<!--
	<Src value="image.fpx" />
	-->
</FPX>
```

Using HTTP Queries

In addition to using XML configuration files you can define FSI parameters by adding a HTTP query to the URL of the FSI Viewer movie clip. In this case parameters have to be provided in the following form:

Parameter1=Value1&Parameter2=Value2…

Characters that are not URL compatible such as "/ ", "% "and "& "within the values need to be URL-encoded.

Providing parameters by HTTP query

```
<Object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.
cab#version=6,0,65,0" width="320" height="240">
      <PARAM NAME="movie" VALUE="/fsi/fsi.swf?cfg=foo&debug=1&MenuAlign=TL">
</Object>
```

Due to the missing possibility to define parameter groups when providing FSI parameters by HTTP query you have to prefix parameters with the name of the parameter group they belong to.

Prefixing parameters with group names

```
XML Syntax:

<FPX>
      <Src value="foo.tif" />
      <ServerType value="fsi" />
</FPX>

Equivalent HTTP Query:

?FPXSrc=foo.tif&FPXServerType=fsi
```

Most parameters belong to the global "options" group. These parameters do not need to be prefix in order to keep HTTP queries as short as possible.

Providing <Options> parameters by query

```
XML Syntax:

<Options>
      <Debug value="true" />
      <MenuAlign value="TL" />
</Options>

HTTP Query equivalent:

?debug=true&MenuAlign=TL
```

The FSI parameter reference provided in this documentation contains the parameter group a parameter belongs to as well as the long parameter including the group names for use in HTTP queries.

Hierarchy of Configuration Parameters

As explained before parameters can be provided in three different ways:

1. The default configuration file "_default.fsi" - located in the same folder as the main viewer file "fsi.swf" - will always be evaluated first
2. You can store image specific parameters in additional *.fsi files
3. You can provide parameters by HTTP query

Parameters specified in multiple ways – for example in the default configuration and by HTTP query – will be evaluated in the following way:

- Parameters defined in configuration files override parameters defined in the default configuration file "_default.fsi".
- Parameters provided by HTTP query override parameters in all configuration files.

The list below shows the hierarchy resulting from the sequential evaluation of the different configuration methods:

Configuration Hierarchy

1. HTTP Query
2. Configuration file (*.fsi)
3. Default configuration file (_default.fsi)

Choosing the adequate Configuration Method

So far you learned that FSI Viewer can be configured by any combination of:

- The default configuration file (_default.fsi)
- Additional configuration files (*.fsi)
- HTTP queries

These configuration options are not only alternate methods. Combining these methods offers the possibility to provide configuration parameters as easily as possible and as dynamically as required. The following considerations might be helpful when deciding which method to choose for a certain parameter.

Default Configuration

You can easily change global parameters that apply to all FSI Viewer instances by editing a single file. This is therefore the recommended configuration method for:

- Global parameters valid for all images, in particular:
    - FPXServerType
    - FPXBase
    - Licence Key
- General configuration options like:
    - Plug-ins
    - MenuAlign, AnimationSpeed and alike

Configuration Files

Separate configuration files should be used for parameters specific to a group of images. You might for example want to setup individual configuration files for different types of FSI Viewer instances, like "small_no_interface.fsi", "large_with_hotspots.fsi" as an example.

Try to avoid adding global parameters like the base path to your imaging server "FPXBase" or alike in individual configuration files. This way you will only need to change the "_default.fsi" configuration file once you need to change the address of your imaging server.

Usually you don`t need to create a configuration file for an individual image, as you can specify the image path or other unique parameters dynamically by HTTP query.

Passing Parameters by HTTP Query

This method is recommended for parameters from databases, parameters specific to an image or FSI Viewer instance, or parameters related to user interaction.
Using server side script or JavaScript you can build HTTP queries to pass parameters to FSI Viewer dynamically.

If you want to display an image with minimal variation in configuration, it is recommended to reuse the same FSI configuration file and to provide varying parameters (like the image path "FPXSrc") by HTTP query.
Try to provide only those parameters by HTTP query that don`t apply to multiple instances of FSI Viewer. Keep in mind that changing a single configuration file for multiple images later on will be far less work compared to editing multiple HTML codes later on.

Providing parameters by HTTP query

```
View 1:
      <PARAM name="movie" VALUE="fsi.swf?cfg=flowers&FPXSrc=rose.fpx">

View 2:
      <PARAM name="movie" VALUE="fsi.swf?cfg=flower&FPXSrc=rose.fpx&NoNav=1">
```

Alphabetical Index of FSI Viewer Parameters

Retrieving Parameters from Imaging Servers

Depending on the type of imaging server in use, FSI Viewer is capable of retrieving image specific parameters provided by the imaging server. The following table shows the parameters that can be retrieved automatically:

	FSI Server	eRez Imaging Server	Iseemedia Imaging Server	True Spectra Imaging Server
FPXWidth	Yes	Yes	Yes	Yes
FPXHeight	Yes	Yes	Yes	Yes
FPXTilesX	Yes	Yes	—	—
FPXTilesY	Yes	Yes	—	—
SceneSets	Yes	Yes	—	—
Image Collections	Yes	Yes	—	—

In which case will these values be retrieved automatically?
FSI Viewer tries to retrieve the parameters FPXWidth and FPXHeight from the server if one of these values has not been defined manually. If the server delivers additional parameters (e.g. FPXTilesX, FPXTilesY) these parameters will only have an effect if they have not been defined manually.

Requirements for Automatic Parameter Supply
If you are using FSI Viewer and FSI or eRez server, you don't need to care about the requirements below as the default setup of FSI Viewer in imaging server ensures communication between server and FSI Viewer within the same domain.

Cross-Domain Access
As Flash movie clips can not access data across domain boundaries by default, FSI Viewer has to be setup to the same domain / IP address as the imaging server.

For imaging servers that integrate into common web servers (FSI server, eRez, TrueSpectra Bridge) this can easily be accomplished by setting up FSI Viewer to a subdirectory of the web server.

For imaging servers delivering images through their own server engine the following solutions apply:

- Setting up FSI Viewer to a web server on the same IP as the imaging server, using a different port (e.g. port 8080)
- Using NeptuneLabs ZoomCache proxy to route FSI Viewer requests
- Setting up FSI Viewer to a subdirectory of the imaging server that is able to deliver standard file types (e.g. the 'servercode' directory of Iseemedia Imaging server

Preventing Security Popup with Flash MX 2004™ plug-ins or above
From Flash MX 2004™ Adobe modified the security model of Flash plug-ins regarding the cross domain policy. To avoid security messages popping up when FSI Viewer tries to retrieve data from the Imaging server you have to add an XML file to the root directory of the imaging server being requested. The file contains information about domains that are allowed to retrieve data from the specific domain and has to be named 'crossdomain.xml'

crossdomain.xml

```
<?xml version="1.0" encoding="iso-8859-1"?>
<cross-domain-policy>
  <allow-access-from domain="*" secure="false" />
</cross-domain-policy>
```

Please refer to the Adobe Flash™ documentation for more information regarding 'cross domain security'.

Basic Parameters

The following parameters are essential and usually need to be specified.
Please refer to the chapter "*Retrieving Parameters Automatically*" for information on how to enable FSI Viewer to retrieve image specific parameter values automatically.

CFG	
Description	Relative path to a configuration file
Syntax	fsi.swf?cfg=[FSI_Name]
Default	—
Context	HTTP query

A relative path to an XML based configuration file "*.fsi". The file extension ".fsi" has to be omitted. You can specify a default base path to all configuration files using the *FSIBase* parameter.

Defining a configuration file

```
<PARAM name="movie" VALUE="fsi.swf?cfg=foobar">
```

In this case the FSI Viewer first searches for the _default.fsi file in the installation folder where the file fsi.swf is located and evaluates the configuration parameters.

The optional parameter "FSIBase" from the _default.fsi file is then used as the path specification in order to search in the correct index for a FSI configuration file with the name "foobar.fsi".

DefaultCFG	
Description	Relative path to an alternate default configuration file
Syntax	fsi.swf?defaultcfg=[filename]
Default	_default
Context	HTTP query only

Relative path to an alternative default configuration file other than "_default.fsi". Only useful as part of an HTTP query. The file extension ".fsi" has to be omitted.

ImageServer	
Description	URL to your FSI Server
Syntax	URL
Default	See description below
Context	<Image>
Version	5.5.0 or higher

The UTL to your FSI Server including the servlet name, e.g.:

```
<ImageServer value=”http://mydomain.tld/fsi/server” />
```

If you do not specify this parameter, FSI Viewer will try to determine the URL based on it’s own URL and the default location of the „viewer“ folder in the FSI Server setup.

Base	
Description	Image Server URL
Syntax	URL
Default	—
Context	<FPX>

Obsolete: Please use parameter ImageServer instead
Domain and path to the imaging server.
All "FPXSrc" parameters containing relative paths (without "http://...") will be extended by the path defined by "FPXBase".

ServerType	
Description	Image Server type
Syntax	String –FSI | eRez | ZIS | TrueSpectra | TrueSpectra Bridge
Default	—
Context	<Image> (or alias <FPX>)
Query	ImageServerType (alias: FPXServerType)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

Please enter the value "fsi" for FSI Server. Please enter the value "erez" for YaWah eRez Imaging Server. For the Zoom Image Server from iSeeMedia please enter the value "ZIS".
For the Image Server from TrueSpectra please enter the value "TrueSpectra" if you are using the stand alone server and "TrueSpectra Bridge" if you are using the TrueSpectra Bridge API.

ServerTemplate	
Description	Real-time template of the imaging server
Syntax	String – template name
Default	fsi
Context	<Image> (or alias <FPX>)
Query	ImageServerTemplate (alias: FPXServerTemplate)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

This parameter applies to FSI Server and eRez Imaging Servers only and specifies the name of a real-time template used when accessing the imaging server.
Using these templates you can specify e.g. the image compression, maximum image dimensions and watermarks.

ImagePath	
Description	Path of the image to display
Syntax	String
Default	—
Context	<Image>
Version	5.5.0 or higher

The path to the image on your FSI Server including the profile name, e.g.:

```
<ImagePath value=”images/samples/foo.jpg” />
```

Src	
Description	URL of the source image to be displayed
Syntax	URL
Default	—
Context	<FPX>
Query	FPXSrc

Obsolete: Please use ImagePath instead
Specify either a relative URL or the absolute URL of the source image on the imaging server. The parameter *FPXBase* can be used to specify a base path to the imaging server.
Please refer to the documentation of your imaging server for details on supported image formats.

Absolute addressing (FSI Imaging Servers)

```
<FPX>
       <Src value="http://www.domain.com/fsi/server?source=project/foo.tif" />
</FPX>
```

Absolute addressing (Iseemedia Imaging Servers)

```
<FPX>
       <Src value="http://www.domain.com/fif=project/foo.fpx" />
</FPX>
```

Partial (relative) addressing

```
<FPX>
       <Src value="project/foobar.fpx" />
</FPX>
```

Width and Height	
Description	Source image dimensions
Syntax	Number in pixels
Default	—
Context	<Image> (or alias <FPX>)
Query	ImageWidth and ImageHeight (or alias: FPXWidth and FPXHeight)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

Required parameters specifying the width and height of the source image.
Please refer to the chapter "*Retrieving Parameters Automatically*" for information on how to automatically retrieve these values from the imaging server.

Width and height specifications for a source image

```
<FPX>
...
       <Width  value="3330" />
       <Height value="4660" />
...
</FPX>
```

Optional Parameters

The following parameters are not required to run FSI Viewer. They can be used to alter the appearance of the FSI Viewer user interface.

Skin	
Description	Filename of FSI Skin to use (without extension)
Syntax	String
Default	—
Context	<Options>

Filename of FSI Skin file to use (omit the file extension). FSI Skins determine the appearance of the user interface and are located in the subdirectory "/skins".
In addition to the default skins that ship with FSI Viewer customized skins are available on demand.
Please note that using unregistered skins activates the evaluation mode.

Intro	
Description	Filename of start animation to use (without extension)
Syntax	String
Default	—
Context	<Options>

Filename of animation to display on startup (omit the file extension).
FSI Viewer ships with a default start animation. Custom intros are available on demand.
Please note that using an unregistered intro activates the evaluation mode.

Debug	
Description	Activate debug mode
Syntax	Bool
Default	false
Context	<Options>

Activate the debug mode providing configuration details, warnings and error messages. It is strongly recommended to activate the debug mode when implementing new images with FSI Viewer.
Please refer to the chapter *Debug Mode* for a detailed description.

FSIBase	
Description	Path to FSI configuration files
Syntax	Relative path
Default	—
Context	<Options>, _default.fsi only

Enter the relative path from FSI Viewer to the image specific configuration files here. If this parameter is specified the path is added to all relative CFG parameters.

Definition of FSIBase in the _default.fsi file

```
<Options>
...
       <FSIBase value="config/" />
...
</Options>
```

Specifying FSI files using the CFG parameter

```
<Object ...>
...
       <PARAM name="movie" VALUE="/fsi/fsi.swf?CFG=imagedir/image1">
...
</Object>
```

The content of the parameter "CFG" will be extended by the parameter "FSIBase" to "config/imagedir/image1.fsi"

FPXBase Parameter

```
Definition of FPXBase within the _default.fsi file:
<FPX>
...
       <Base value="http://domain.com/fsi/server?source=" />
...
</FPX>

Specification of the FPXSrc within configuration files:

1) FPXBase parameter is used for relative addressing:
<FPX>
...
       <SRC value="folder/image.tif" />
...
</FPX>

The resulting address is:
http://domain.com/fsi/server?source=folder/image.tif

2) FPXBase is not used for absolute addressing:
<FPX>
...
       <SRC value="http://domain.com/fsi/server?source=project/image.tif" />
...
</FPX>
```

Display a thin border around FSI Viewer. The default value is "false" displaying no border. Enter one 6-digit hexadecimal color (e.g. "FF0000") for a solid border. If you specify two hexadecimal colors (e.g. "FF0000, 00FF00") the top left border will use the first color while the bottom left border uses the second color.

CropRect	
Description	Defines a crop rectangle
Default	0,0,1,1
Context	<Options>
Version	5.0.0

Crop rectangle (left, top, right, bottom) as 0..1 floating point values. Default: 0,0,1,1

MenuAlign	
Description	Alignment of the menu bar
Syntax	String
Default	version 1-4: TL, from version 5: BC
Context	<Options>
Version	The "center" option "C" is available from version 5.0.0

Possible values:

TL	(top-left)	TC	(top-center)	TR	(top-right)
BL	(bottom-left)	BC	(bottom-center)	BR	(bottom-right)

Language	
Description	Interface language
Syntax	String
Default	english
Context	<Options>
Version	FSI Viewer 2.0 or higher / FSI Showcase 2.15 or higher

Defines the language of tool tips for the user interface.
Possible values depend on the XML based language files located in the sub-index "/languages" of the FSI Viewer setup path. To specify a language, please enter the filename without file extension, e.g. "german".
You can modify existing or add your own *language files*.

ImagePanButtons	
Description	Show or hide pan buttons on the image
Syntax	Boolean
Default	false
Context	<Options>
Version	4.0.0 and above

Show or hide the optional pan buttons on the image. If the user points at a pan button, the image pans into the direction indicated. The buttons automatically disappear if the edge of the image has been reached.

DetailBuffer	
Description	Enable or disable the detail buffer
Syntax	Boolean
Default	true
Context	<Options>
Version	3.0.0 and above

Enabling the detail buffer keeps previously loaded image details visible when zooming in or out instead of displaying the low resolution preview image while loading additional image details. This way zoom transitions appear smoother.

NoNav	
Description	Hide the navigator window
Syntax	Boolean
Default	false
Context	<Options>

Show or hide the navigator window displaying the miniature image and the current image section.

NavWidth and NavHeight	
Description	Maximum size of the navigator window
Syntax	Number in percent or pixel
Default	80
Context	<Options>
Version	3.0.0 and above

Defines the maximum width and height of the image inside the navigator window. The value can be defined absolutely or in percent of the viewer size. Use parameter *NoNav* to hide the navigator window.

NavFrameColor	
Description	Color of the selection frame inside the navigator window
Syntax	HexColor
Default	FF0000 (depending on the skin)
Context	<Options>
Version	3.0.0 and above

Specify a 6-digit HexColor to change the color of the selection frame inside the navigator window, e.g. "0000FF" for blue.

NavMaskColor	
Description	Color of the area around the selection frame of the navigator window
Syntax	HexColor
Default	depends on the skin
Context	<Options>
Version	3.0.0 and above

Specify a 6-digit hexadecimal color to change the color of the area outside the selection frame of the navigator window, e.g. "FFFFFF" for white.

NavMaskAlpha	
Description	Opacity of the area around the selection frame of the navigator window
Syntax	Number (0..100)
Default	depends on the skin
Context	<Options>
Version	3.0.0 and above

Specify the opacity of the area outside the selection frame of the navigator window. Possible values are integer values from "0" (invisible) to "100" (opaque).

InitialView	
Description	Initial image segment
Syntax	SceneSet, Scene [left, top ,right, bottom, rotation]
Default	1,1,0,0,1,1,0
Context	<Options>

Image section (and rotation) to display on startup.
The first two parameters are required as they specify the scene set and the scene. For 2D images both values default to 1. The other parameters are optional and specify the image segment and the rotation around the z-axis. Please use the plug-in *SelectFrame* to easily retrieve valid values for this parameter.

InitialView

```
<Options>
...
       <InitialView value="1,5,0,0,0.25,0.25,90" />
...
</Options>
```

The example displays the above left quarter of the 5th scene in the 1st scene set rotated 90° to the right.

InitialViewPersistent	
Description	Keep the InitialView as default view
Syntax	Boolean
Default	false
Context	<Options>
Version	2.0.0 or higher

If this parameter is activated, clicking the "Reset" button will display the image section defined by the InitialView parameter instead of the entire image. If this parameter is set to "false", the initial view will be displayed on startup, but pressing the Reset button will reset the view to the entire image.

InitialAction	
Description	Action on startup
Syntax	String
Default	—
Context	<Options>

Specifies an action to execute on startup. The action is repeated until the user presses a button or key.
While the action is being executed no image details are loaded.

Possible values:

NextScene	(rotate around the y-axis to the right,	3D only)
PreviousScene	(rotate around the y-axis to the left,	3D only)
NextSceneSet	(rotate around the x-axis to the right,	3D only)
PreviousSceneSet	(rotate around the x-axis to the left,	3D only)
RotateRight	(rotate around the z-axis to the right,	3D only)
RotateLeft	(rotate around the z-axis to the left,	3D only)
ZoomIn	(zoom in)	
ZoomOut	(zoom out)	

InitialActionDelay	
Description	Speed of the initial action
Syntax	Number
Default	3
Context	<Options>

Specifies the speed of the action specified by the InitialAction parameter.
Possible values are integer values greater than 0.
0 highest speed
>1 slower

InitialActionPersistent	
Description	Repetition of initial action
Syntax	Boolean
Default	false
Context	<Options>
Version	2.2.0 or higher

Initial action restarts on each ‘reset‘ command (button or keyboard) if set to true.
Otherwise the action only performs on startup.

HelpURL	
Description	URL of custom help page
Syntax	String
Default	http://help.fsi-viewer.com
Context	<Options>
Version	1.3.0 or higher

If you want to create a custom help page for your FSI Viewer enter the complete (absolute) URL to this page using this parameter.

HelpURLTarget	
Description	Target (frame) for custom help page
Syntax	String
Default	_blank
Context	<Options>
Version	1.3.0 or higher

Using this parameter you can define the target name of the browser window if a user opens the help page. The default value "_blank" opens a new browser window.

Initial MouseMode	
Description	Mouse mode on startup
Syntax	Number
Default	0
Context	<Options>

Specifies the selected mouse mode on startup.
Possible values:
0 (Zoom)
1 (Pan)
2 (Rotate 3D)
3 (Rotate 2D)

Some plug-ins provide additional mouse modes. Please refer to the corresponding *plug-in reference* for details.

EnableZoomInPanMode	
Description	Enable zooming in pan mode
Syntax	Boolean
Default	false
Context	<Options>
Version	4.1.2

Set this parameter to true if you want to enable the user to zoom by clicking the image while mouse mode "pan" is enabled.

HandCursor	
Description	Shape of the mouse cursor
Syntax	Boolean
Default	true
Context	<Options>

If you set this parameter to "false" the default arrow cursor will be used instead of the hand cursor when hovering over the image or the navigator window.

ZoomLimit	
Description	Maximum magnification
Syntax	Number
Default	100
Context	<Options>

Defines the maximum magnification in percent relative to the source image.
The default value "100" permits magnification to the physical resolution of the source image. Possible values are "1" to "1000".

ZoomLimitMin	
Description	Minimum magnification
Syntax	Number
Default	100
Context	<Options>

Defines the minimum magnification in percent relative to the initial image.
The default value "100" permits magnification to the size of the Viewer. Smaller values permit magnifications smaller than the viewer size. Possible values are "1" to "100".

ExtendedViewport	
Description	Extend the size of the viewport
Syntax	Boolean
Default	false
Context	<Options>

With this option enabled you can pan the borders of the image to the center of the view port. If you disable this option the panning range is smaller so that the user can not pan the image borders to the center of the view port.

StrictViewport	
Description	Strict viewport allowing minimum image panning only
Syntax	Boolean
Default	false
Context	<Options>
Version	4.1.0 or above

With this option enabled the pan range for the image is limited to the minimum required to view the entire image. This option can be useful if you want the user to pan in one direction only, because the image width matches the view port width or height (e.g. when using FSI Pages with "autozoom=fitWidth" option).

MagnifySmallImages	
Description	Magnify images smaller than the viewport
Syntax	boolean
Default	false
Context	<Options>
Version	4.1.0 or above

Usually FSI Pages 4.1.0 or above does not magnify small images to fit the view port. You can enable this option in case you want small images to fit the view port even if the source image is smaller than the view port.

Animation	
Description	Animation quality
Syntax	Boolean
Default	true
Context	<Options>

Animate zoom and rotate actions (true) or zoom and rotate in a single step (false)

AnimationType	
Description	Defines type of zoom animation
Syntax	Number
Default	3
Context	<Options>
Version	5.0.0

Defines the type of the zoom animation (acceleration on start or end, bounce, etc.) Possible values are 0 to 6.

AnimationSpeed	
Description	Animation speed
Syntax	Number
Default	50
Context	<Options>

Specifies the speed of the zoom and scene animations. Possible values range from 1 (very slow) to 100 (very fast). See parameter "*Animation*" to disable animated motion.

HideUI	
Description	Hide user interface
Syntax	Boolean
Default	false
Context	<Options>

Hide the menu bar on startup. The user has to click the "Show Menu" button to show the user interface. This option is especially useful when an image is meant to look like an ordinary image or if there are many FSI Viewers on a single web page.

RoundedSkinCorners	
Description	Defines round skin corners
Default	—
Context	<Options>
Version	5.0.0

Defines the radius of the rounded viewer corners. Possible values range from 0 (no rounded corners) to 99.

UISwitchable	
Description	Enable hiding/showing the user interface
Syntax	Boolean
Default	true
Context	<Options>
Version	3.2.2 and above

Specifies whether the user is able to show or hide the user interface by clicking the corresponding button or pressing "0" on the num pad. The "Show Menu" button is hidden if this option is set to "false".

ViewerBorder	
Description	Defines the color of the viewer borders
Default	—
Context	<Options>
Version	5.0.0

Color of the border as a hexadecimal color value, e.g. FF0000 for red. Use "false" for no border.

NoImageBlend	
Description	Subsequently loaded image details display without transition
Syntax	Boolean
Default	false
Context	<Options>
Version	1.2.0 or higher

If this parameter is activated, subsequently loaded image details are displayed abruptly without transition.

Effects	
Description:	Image manipulation parameters for all images
Syntax:	String
Default:	—
Context:	<Options>
Version:	1.3.0 or higher

The effects parameter can be used to pass image manipulation parameters to the imaging server. The range of available parameters depends on the imaging server being used. Typical manipulation parameters include the image compression ("quality" or "qlt") and image sharpening/blurring ("sharpen" or "ftr"). If no effect is specified, the default settings of the server apply.
Please refer to your imaging server for a list of available parameters.

Example "quality"
The "quality" or "qlt" parameter specifies the JPEG compression of images delivered by the imaging server.

Practical values range from 50 to max. 90

quality=5

```
<effects value="quality=5" />
```

quality=90

```
<effects value="quality=90" />
```

Example "sharpen"
The "sharpen" or "ftr" parameter blurs (negative values) or sharpens (positive values) images delivered by the imaging server.

sharpen=-300 (blur)

```
<effects value="sharpen=-300" />
```

sharpen=150

```
<effects value="sharpen=150" />
```

Other effects change, for example, the color saturation, hue or contrast.
Please refer to your imaging server documentation for a complete description of available image effect commands.

brightness=10

```
<effects value="brightness=10" />
```

brightness=-10

```
effects value="brightness=-10"/>
```

The format of the effects parameter is similar to a HTTP query. You can combine different effects by concatenating the "parameter=value" pairs with an ampersand (&) character.
<effects value="quality=65&sharpen=30"/>
When passing effects by HTTP query the value of the effects parameter has to be provided as an *url-encoded* string (e.g. "%26" instead of "&" and "%3D" instead of "=").

...fsi.swf?effects=quality%3D65%26sharpen%3D30

In general FSI Viewer passes the value of the effects parameter through to the imaging server. Therefore, using the effects parameter, you can pass through all parameters your imaging server offers for static image manipulation.

For example with eRez imaging server you can force the use of a real-time template by passing the "tmp=" parameter:

<effects value="tmp=fsi"/>

Applying Effects to Specific Images Only

While the effects parameter holds image manipulation parameters for all images you can also apply effects to specific types of images in FSI Viewer (see "NavEffects", "SceneEffects" and "TileEffects").

For example it is sometimes useful to sharpen the images inside the navigator window as small images tend to appear a little blurry. To do so you can add a corresponding "NavEffects" parameter.
Sharpening Images in the navigator window:

<NavEffects value="effects=Sharpen(30)"/> (FSI Server)
<NavEffects value="sharpen=35"/> (eRez Imaging Server)
<NavEffects value="ftr=5"/> (Iseemedia Zoom Image Server)

NavEffects	
Description	Image manipulation parameters applied to the navigator window
Syntax	String
Default	—
Context	<Options>
Version	3.1.1 or higher

Applies image manipulation effects like "sharpen" or "quality" to all images inside the navigator window. Please refer to "effects" parameter for a detailed description.

SceneEffects	
Description	Image manipulation parameters applied to preview images
Syntax	String
Default	—
Context	<Options>
Version	3.1.1 or higher

Applies image manipulation effects like "sharpen" or "quality" to all un-tiled images in the viewer. Please refer to "effects" parameter for a detailed description.

TileEffects	
Description	Image manipulation parameters applied to image tiles
Syntax	String
Default	—
Context	<Options>
Version	3.1.1 or higher

Applies image manipulation effects like "sharpen" or "quality" to all image tiles in the main view area. Please refer to "effects" parameter for a detailed description.

ZoomCache	
Description	URL of the ZoomCache
Syntax	URL
Default	—
Context	<Options>

If you use NeptuneLabs ZoomCache, please specify the URL of the ZoomCache server here. You can receive this URL and a ZoomCache ID from your image hosting provider.

ZoomCacheID	
Description	ID of the ZoomCache access profile
Syntax	String
Default	—
Context	<Options>

When using NeptuneLabs ZoomCache enter the name of the access profile for your images here. You may have several profiles i.e. if you use watermarking or different imaging servers.

IgnoreQuery	
Description	Ignore all parameters passed via query (except **c**fg)
Syntax	Boolean
Default	false
Context	<Options>

Ignore all parameters passed via HTTP-query except for the "cfg" parameter.
This way you can entirely disable modifications to the FSI configuration by query.

IgnoreQueryParameters	
Description	List of parameters to ignore in queries
Syntax	String
Default	—
Context	<Options>
Version	5.0.4 or higher

A comma separated list of parameter names that should be ignored if specified in the HTTP query.
Example: <IgnoreQueryParameters value="language, skin" /> would prevent specifying the "skin" and "language" by HTTP query.

Please note:

- "DefaultCfg", "Cfg" and "Debug" parameters cannot be disabled using this parameter
- this parameter itself cannot by specified by query

MaxZoomLevel	
Description	Load image details at maximum zoom
Syntax	Boolean
Default	true
Context	<Options>
Version	3.2.1 or higher

By default FSI Viewer adds a zoom level for the maximum magnification – this is FSI Viewer loads image details when reaching the maximum magnification. For extremely high resolution 2D images exceeding 32,000 pixels in width or height you can set this value to "false" to prevent display problems at the maximum magnification.

TiltModeRelative	
Description	Use absolute or relative tilt mode
Syntax	Boolean
Default	true
Context	<Options>
Version	3.6.0 or higher

True: Rotate the image (2D) around the center of the view port depending on the mouse position and distance to the center. False: Rotate the image (2D) clockwise moving the mouse right and counterclockwise moving the mouse left.

TileSizeX and TileSizeY	
Description	Size of image detail tiles
Syntax	Number in pixel or in percent of the viewer size
Default	50%
Context	<Options>
Version	2.3.0 or higher

Specifies the size of image details (image tiles). The value can be defined relative to the viewer size (e.g. "25%") or using absolute pixel values (e.g. "256"). The default value of 50% means that each image tile is one quarter in size of the entire viewer. This way a maximum of 9 image tiles is required to display the selected image section.
Please do not change this value unless you are certain that this is necessary and you are aware of possible consequences:

- Defining smaller image tiles reduces the overall viewer performance and increases the number of requests and thus the imaging server load.
- Defining larger image tiles might lead to increased traffic and decreased download performance as more image data than actually required for a certain image section will be loaded.

InfoTemplate	
Description	Configuration template to retrieve image info from server
Syntax	String
Default	fsi/info.xml
Context	<Image> (or alias <FPX>)
Query	ImageInfoTemplate (or alias FPXInfoTemplate)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

Specifies the configuration template used to retrieve image specific data from FSI or eRez imaging servers. You might want to specify a custom template if you want to retrieve IPTC data for all images.

MetaDataQueryParameters	
Description	HTTP Query to append to meta data requests
Syntax	Query
Default	—
Context	<Options>
Version	4.1.3 or higher

An optional query string of the form "a=1&b=2&c=3…" to append to meta data request to FSI or eRez Imaging server. These query parameters will be appended to info and list requests passed to FSI or eRez server.

ConcurrentRequests	
Description	Number of simultaneous image requests
Syntax	Number
Default	2
Context	<Options>
Version	2.3.0 or higher

Specifies the number of image details (image tiles) being requested simultaneously.

Please do not change this value unless you are certain that this is necessary and you are aware of possible consequences:

- Increasing this value might lead to increased imaging server load
- Decreasing the value will decrease the download performance

UseIntermediateZoomLevels	
Description	Use more zoom levels than contained in the source image
Syntax	Boolean
Default	true
Context	<Options>
Version	4.0.0 or higher

Images with resolution levels contain the same image in different resolutions. The size of the next level is twice the size of the preceding level. If you disable this option FSI Viewer accesses these resolution levels only. Otherwise image data will be retrieved more often.

UniqueID	
Description	Unique Identifier for plug-ins
Syntax	String (see description)
Default	—
Context	<Options>
Version	3.6.0 or higher

A unique string identifying the image / image collection displayed in FSI Viewer.
You can for example use the path of the image or the name of a catalog. Do not use the following characters in UniqueIDs: [Space] ~ % & \ ; : " ` , < > ? #
The *Notepad plug-in* for example requires a UniqueID value.

MouseVelocity	
Description	Continue mouse actions after mouse up
Syntax	Boolean
Default	false
Context	<Options>
Version	5.0.0 or higher

If you enable this parameter mouse actions like "pan" will be continued depending on the speed you moved the mouse.

The higher the drag speed the longer the action will be continued.
This option is especially useful for 3D views. With this parameter enabled and mouse mode set to "rotate 3D" the object will continue to spin rather than stopping immediately after mouse up. Additionally a very fast spin of a 3D view will add a blur effect to the object.

Editing or Creating Language Files

In case you want to edit tool tips for existing languages or add support for additional languages you can edit or create language files in the subdirectory "/languages" of your FSI Viewer setup directory.

Each language file contains a list of phrases used in FSI Viewer user interface.
The language files are XML files and need to be saved in Unicode (utf-8) format.
The attribute "id" specifies the context of each phrase. Please note that you need to use XML entities for characters not allowed in XML files, like e.g.

&amp;	for "&"
&lt;	for "<"
&gt;	for ">"
&quot;	for "
&apos;	for ,

You can use *basic HTML tags* to apply text formatting to the tool tips.

Excerpt from a language file

```
<Data language="english" color="000000" font="_sans" offsetTop="0" offsetLeft="0"
offsetWidth="0" offsetHeight="0">
       <tip id="ZoomOut"><b>Zoom Out</b><br/>Click to decrease magnification</tip>
       <tip id="ZoomIn"><b>Zoom In</b><br/>Click to increase magnification</tip>
       <tip id="Downloading Data">Downloading:</tip>
[...]
</Data>
```

You can duplicate existing language files or create new files as needed.
To use custom language files, please use the "Language" parameter and set the parameter value to the filename without the ".xml" extension.

NeptuneLabs FSI Viewer

Virtual 3D Presentations

Preparing 3D Presentations

In addition to zoomable 2D images FSI Viewer can be used to display virtual 3D presentations of an object. This way you can provide a photorealistic, zoomable and rotateable view of an object. Creating 3D views requires multiple images of the same object from different angles.

There are two different methods to setup virtual 3D views:
Using multiple source images
Using a single, tiled source image

Method A - Using Multiple Source Images

This is the recommended way to setup virtual 3D views.
Upload the images belonging to a 3D View to a subdirectory of your imaging server. Create a configuration file (e.g. "my3dview.fsi") and add the list of source images belonging to this view like in the sample as shown:

Configuration for a 3D presentation as displayed in Figure A on page 53

```
<?xml version="1.0" encoding="$charset"?>
<fsi_parameter>

	<images3d prefix="my3Dimage/image_" suffix=".tif">
		<image src="01" />
		<image src="02" />
		<image src="03" />
		<image src="04" />
		<image src="05" />
		<image src="06" />
	</images3d>

</fsi_parameter>
```

Please note that the sequence of the <image> nodes defines the sequence of the 3D scenes. You can use the parameter "SceneSets" to modify the sequence by configuration later on.

You can either enter the complete path for each image or use the prefix and suffix attributes to have FSI Viewer assemble the file names from prefix + images.src + suffix

In the example above, the image path for the first image is therefore: my3dimage/image_01.tif

Besides adding the <images3d> node to your configuration file you can alternatively load an external file containing the image list. This can be useful if you have a server side script or database containing the image paths for 3D views. In this case you only need to specify the images3dURL parameter which should contain the URL to the image list to retrieve.

Note for FSI Server Users
If you are using Imaging server you can have FSI Server setup the image list for you automatically based on a directory. In this case you can choose "Publish as FSI Viewer 3D" from a directory. In this case the resulting code snip will contain the parameter „Images3DUrl" with the value of the directory path, e.g.:
`[…]&images3durl=images%2Fmy3DImages[…]`

For code snips created for eRez server (including the eRez commands) FSI Viewer Version 5.5 and above extracts the folder path from the „images3durl" parameter value to make the parameter work with FSI Server.

Note for eRez Users
If you are using Imaging server you can have eRez setup the image list for you automatically based on a directory or search query. In this case you can choose "Publish to Web" => "Publish as FSI Viewer 3D" from either a directory or search result. In this case the resulting code snip will contain the parameter „Images3DUrl" with parameters to retrieve the image list, e.g.:

```
[…]&images3durl= %3Fcmd%3Dlist%26src%3Dimages%2Fmy3dImages%26vtl%3Dfsi%2Fimage_list3d.xml
%26page%3D1%26rows%3D16%26cols%3D16[…]
```

Method B - Using a single, tiled source image

In this case you need to prepare the source image in the way, that it contains all required images as tiles of equal dimension. Please see the figure below which shows a so prepared source image with 12 image tiles.

The number of horizontal and vertical image tiles must be specified using the parameters *TilesX and TilesY*.

Figure A displays a source image containing 12 tiles representing a rotation of 360° around the Y-axis in steps of 30°. In this case you simply add the parameters TilesX=3 and TilesY=4 to your FSI configuration.

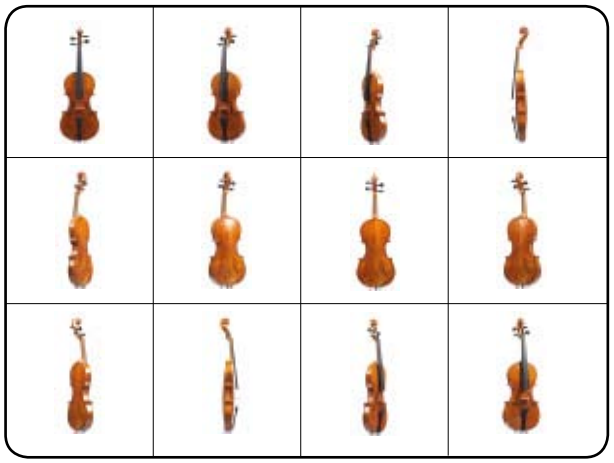


Fig A: Rotation around Y-axis in 30° steps
4 x 3 = 12 tiles

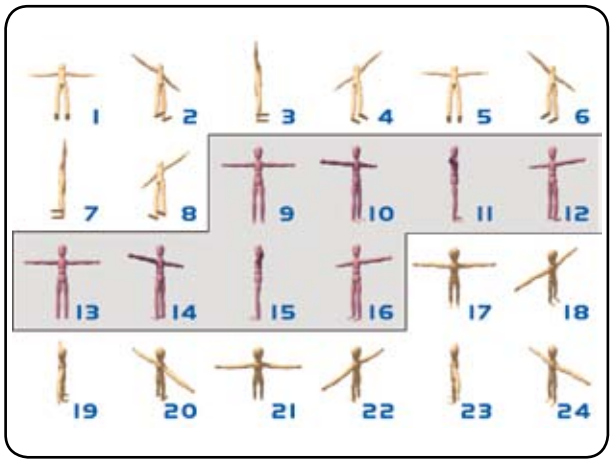


Fig B: Rotation around Y-axis in 45° steps and partial rotation around X-axis in 15° steps
6 x 4 = 24 tiles in 3 scene sets

Example – Configuration for a 3D presentation as displayed in Figure A

```
<?xml version="1.0" encoding="$charset"?>
<FPX>
       ...
       <TilesX value="3" />
       <TilesY value="4" />
       ...
</FPX>
```

Figure B shows a more complex 3D view representing a rotation of 360° around the X-axis in 45° steps and an additional rotation of +/- 15° around the Y-axis.

To present rotations around more than one axis or to present an object in different states you need to define *SceneSets*.

A Scene Set describes a sequence of individual images.
The user can select individual images of a scene set using the keys "7" or "9" in the num pad or using the mouse mode "rotate" moving the mouse horizontally.
The user can additionally select a scene set using the keys "/" and "*" in the num pad or using the mouse mode rotate moving the mouse vertically.

If the TilesX or TilesY parameters are defined for a source image, but no scene set is defined, a scene set containing all the scenes from left to right, top to bottom will be created by default. Referring to Figure B this would be

```
<SceneSets value="1-24" />
```

As the presentation of Figure B consists of 6 by 4 individual images while 8 images represent a complete rotation around the Y-axis the definition of the following 3 scene sets is required:

Image Tiles	Y-Axis	X-Axis
1 – 8	0 to 360°	- 15°
9 – 16	0 to 360°	0°
17 – 24	0 to 360°	+15 °

The required configuration to rotate the object around the Y-axis moving the mouse horizontally and rotating the object around the X-axis moving the mouse vertically looks like this:

Configuration for a 3D view as displayed in Figure B

```
<FPX>
       ...
       <TilesX value="6" />
       <TilesY value="4" />
       ...
</FPX>

<Options>
...
       <SceneSets value="1-8;9-16;17-24" />
...
</Options>
```

You can use the SceneSet parameter to specify the image sequence independent of the way you setup the 3D view (multiple source images or a tiled source image). Please refer to the description of the *SceneSets* parameter for details on the different ways of defining images contained in a scene set.

Tip: If you are not sure about the number of images in your tiled source image in the x and y direction, try entering the value 1 for TilesX and TilesY. In the viewer you will then see the entire source image structure.

Parameters for Virtual 3D Presentations

The following FSI configuration parameters apply to virtual 3D presentations only.

Parameters for multiple source images (Method A) only:

Images3D	
Description	Node containing the list of source images
Syntax	XML node
Default	—
Context	ROOT node like <FPX> or <Options>
Query	n/a

Add this node to the top level of a configuration file to list the individual images for a 3D presentation. You can add the attributes prefix and suffix instead of providing the complete path for each image individually.

Configuration for a 3D presentation as displayed in Figure A

```
<?xml version="1.0" encoding="$charset"?>
<fsi_parameter>

      <Images3d prefix="my3Dimage/image_" suffix=".tif">
            <image src="01" />
            <image src="02" />
            <image src="03" />
            <image src="04" />
            <image src="05" />
            <image src="06" />
      </Images3d>

</fsi_parameter>
```

Images3DURL	
Description	URL to an XML file containing a 3D image list
Syntax	URL
Default	—
Context	<options>

URL to an external XML file containing the image lists for 3D presentations as explained above. You might want to use the parameter Images3DURLBase in your _default.fsi configuration file to specify the URL base for all 3D image list URLs.

Images3DPrefix, Images3DSuffix	
Description	Prefix / Suffix to add to 3D image URLs
Syntax	String
Default	—
Context	<Options>
Version	4.0.7

Prefix or suffix to add to the <image src=""> values. You might want to use these parameters to pass a prefix or suffix by HTTP query rather than using the corresponding "prefix" and "suffix" attributes in the <images3d> node.

Parameters for tiled source images (Method B) only:

TilesX and TilesY	
Description	Number of individual images for 3D presentations
Syntax	Number
Default	1
Context	<Image> (or alias <FPX>)
Query	ImageTilesX and ImageTilesY (or alias FPXTilesX and FPXTilesY)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

Number of individual images contained in the source image in x- and y- direction respectively.
Example: 3D image containing 6 x 4 tiles arranged in 3 scene sets.

Even with ‘asymmetrical‘ scene orders - like in the example to the left -the number of tiles would be 6 in the x direction and 4 in the y direction.

Definition of 3 scene sets containing 6 x 4 = 24 tiles

```
<FPX>
...
       <TilesX value="6" />
       <TilesY value="4" />
...
</FPX>
<Options>
...
       <SceneSets value="1-8,9-16,17-24" />
...
</Options>
```

Parameters for all 3D presentations (Method A or Method B):

SceneSets	
Description	Scene sequence definition (3D)
Syntax	String
Default	1 – n tiles
Context	<Options>

Using this parameter you can define scene sets for 3D views.
Scene sets define the order of scenes to be displayed.
The default is a single scene set containing all scenes (image tiles) from scene 1 to the last scene image. The default sequence is left to right, top to bottom in a tile image and the sequence of the <image> nodes for 3D views based on individual source images.
In the viewer scenes can be selected by moving the mouse horizontally and scene sets can be selected by moving the mouse vertically.

Syntax:
Individual sets have to be separated with ";" (semicolon).
Scenes within the sets are separated with "," (comma) and specify the 1-based index of the image from left to right, top to bottom up to the total number of scenes (TilesX x TilesY or individual source images).

Instead of individual scenes you can also specify scene ranges, e.g. "5-10".
If you leave out the start or the end of a scene range, 1 or respectively the last scene will be assumed in this case. If you specify a range using "->" the scenes from the first number to the last scene image plus the first scene image up to the second number will be assumed. The example below illustrates the syntax:

Definition of multiple scenes and scenesets

```
<Options>
...
       <SceneSets value="2,3,5-10;10-1;5-;8->3" />
...
</Options>
```

The example defines 4 scene sets in total. Assumption: The source image is created from 10 individual images (scenes), 5 in a row, two rows.
This leads to the following image order for the individual scenes:

Set 1 - Scenes: 2,3,5,6,7,8,9,10
Set 2 - Scenes: 10,9,8,7,6,5,4,3,2,1
Set 3 - Scenes: 5,6,7,8,9,10
Set 4 - Scenes: 8,9,10,1,2,3

You can review the resulting scene sets in the "Info" section of the Debug Window.

ScenePreload	
Description	Load scenes in high resolution (3D)
Syntax	Boolean
Default	false
Context	<Options>

If this parameter is activated (default) all scenes will be loaded at high resolution on startup. Otherwise preview images will be loaded with at a lower resolution. This parameter can significantly influence loading times and traffic volumes. Setting this parameter to false leads to faster startup times and high resolution scene images will be loaded as soon as they need to be displayed.

TilePaddingX and TilePaddingY	
Description	Crops each scene image by a given number of pixel
Syntax	Number (pixel) or percent
Default	0
Context	<Image> (or alias <FPX>)
Query	ImageTilePaddingX and ImageTilePaddingY (alias: FPXTilePaddingX and FPXTilePaddingY)
Version	5.5.0 or higher (2.1.0 or higher for alias <FPX>)

Specifies a positive number of pixels that each source scene image (2D and 3D) will be cropped by. This parameter can be used to eliminate image margins especially with 3D objects containing polychrome backgrounds.
You can specify the padding either absolutely as a number of pixel or in percent of the source image.

NoSceneAnimation	
Description	No SceneSet animations (3D)
Syntax	Boolean
Default	false
Context	<Options>

Disable animations between scenes and scene sets. This can be useful if different scenes represent different states of an object instead of a rotation.

NoSetLoop	
Description	No 360° rotation around the y-axis (3D)
Syntax	Boolean
Default	false
Context	<Options>

The first and the last scene set will be considered ending points. (E.g. for partial rotations < 360°).
Please note: The blur animation is turned off if NoSetLoop is activated.

NoSceneLoop	
Description	No 360° rotation around the x-axis (3D)
Syntax	Boolean
Default	false
Context	<Options>

The first and last scene in the each scene set will be considered ending points. (E.g. for partial rotations < 360°).
Please note: The blur animation is turned off if NoSceneLoop is activated.

NeptuneLabs FSI Viewer

Debug Mode

Enabling the Debug Mode

The debug mode can be enabled or disabled by passing the "Debug" parameter either by HTTP query or in an XML configuration file.

Please use "debug=1" to enable the debug mode and "debug=0" (or omit the debug parameter) to disable the debug mode.
To disable the debug mode permanently you can safely delete the "/debug" directory of your FSI setup directory.

Example:
If FSI Viewer does not work or look as expected in an HTML page you can quickly debug FSI Viewer as follows:

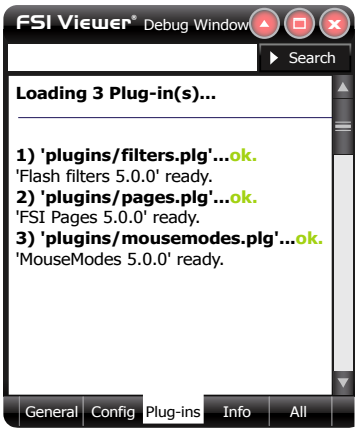
- View the source code of the HTML document and search for the FSI Viewer HTML Code
- Copy the entire FSI Viewer URL (http://...fsi.swf?....) to the clipboard
- Paste the URL to the address bar of your browser appending "**`&debug=1`**"

Using the Debug Window

After activating the debug mode an output window is available presenting information about the initialization progress, configuration parameters, plug-ins and image information.

You can select different debug sections by clicking the corresponding tab at the bottom of the window.

The window can be moved by dragging the window title bar and you can show or hide the window by pressing "d".



The section "All" contains the entire debug output.
Please provide the content of this section when requesting support.

Enabling the debug mode additionally prevents the browser from caching FSI configuration files. This allows you to easily test changes to your FSI configuration files without having to delete the browser cache each time. You can search each section by entering a keyword to the search input at the top. Pressing the "search" button multiple times will cycle through all matches.

General section

Provides general information about the initialization progress. If the viewer does not display an image you should be able to track down the error cause by determining the step where the initialization process stops.

For example if the last output is "Loading Skin and Module" the skin file is most likely missing or corrupt.

If the Viewer runs in evaluation mode you will be presented the reason why evaluating the licence key failed here. Please refer to the plug-in section for a list of active plug-ins and to the Info section for the domain evaluated by FSI Viewer.

Config section

Displays information on the configuration parsing process, the resulting configuration and possible configuration errors.

If the viewer loads successfully, but the configuration does not look as expected you should be able to track the reason in this section. This section lists all recognized parameters resulting from the 3 different configuration methods (default configuration, configuration file and query) and displays tips regarding false parameter values as well as unknown parameters.

Plug-ins section

Provides information on loaded FSI Plug-ins and configuration errors related to plug-ins. Additionally you can review all configuration parameters of the active plug-ins.

Info section

Displays version and licence information, general information related to the viewer instance and runtime information on the image being displayed.

Press "i" to refresh the image section being displayed after zooming, panning or rotating the image.

Debugging Image Access Problems

You can configure the debug window to debug image access as well. To do so you need to set the global parameter "DebugImageAccess" to true.
When enabling the debug mode FSI Viewer outputs error information if problems retrieving images from the imaging server occur.

Please note that this feature requires the imaging server to reside in the same domain as FSI Viewer or using Flash plug-in version 7 together with a valid

"*crossdomain.xml*" file copied to the imaging servers root directory.

NeptuneLabs FSI Viewer

Automated Implementation of Images

Rather than creating individual configuration files for each image it is recommended to provide FSI parameters via HTTP queries appended to the FSI Viewer URL when implementing large numbers of images.
As some browsers require the obsolete <embed> tag, queries have to be added to both the <object> and the <embed> tag. (*HTML-Source Code*).

FSI Viewer is able to request image specific parameters like the source image dimension (parameters "FPXWidth" and "FPXHeight") from your imaging server (*Retrieving Parameters Automatically)*. Using this feature reduces the required work to dynamically providing the path to the source image (parameter "FPXSrc").

It is recommended to use server side script like ASP, PHP, JSP, Perl, etc. when implementing large numbers of images. All subsequent code examples in this section refer to PHP, but can be easily adapted to any other server side scripting language.

Steps to be taken when using large numbers of images:

1. **Defining default parameters**
   Be sure to provide as many recurring parameters as possible in the _default.fsi file.
   This way you don't have to pass these parameters via query.
   The most important parameter is FPXBase, as this parameter enables you to subsequently define FPXSrc parameters using relative addressing.
   If there are some images that require different parameter values than defined in the _default.fsi file, you can still overwrite the default setting by passing the value via query.

2. **Creating image specific URLs**
   Be sure to have image specific data available for server scripting by using a database, your content management system, or any other appropriate source.
   You will at least need to provide the FPXSrc parameter from your data source.
   After collecting the image specific data from your data source you have to build a standard HTTP query string (RFC1738) containing the data using server side scripting (*Using HTTP Queries*).

Generating FSI queries (fetch_fpx_image.php)

```
// Retrieve FPX properties for picture 1 from table "FPX_Images"
$result = mysql_query('select FPXSrc,FPXHeight,FPXWidth,TilesX,TilesY from FPX_Images
where PictureID=1');

if ($result) {

       // Fetch SQL data
       $fpx_parameter=mysql_fetch_assoc($result);

       if ($fpx_parameter){

              // RFC1738 parameter encoding
              foreach ($fpx_parameter as $fpx_key => $fpx_val) {
                     $query_collection[]= $fpx_key.'='.rawurlencode($fpx_val);
              }

              // Make one large query string
              $fsi_query=implode('&', $query_collection);

              // Add Query to FSI Viewer URL
              $fsi_url='fsi.swf?'.$fsi_query;
       }
}
```

3. **Creating the HTML-Code dynamically**
   Create a variable that contains the <object> and <embed> tag with all variables provided as script variables. Replace all variables inside the template using server side scripting.

   You will usually need to replace the following 4 variables:

   - URL (see above: 2. Creating image specific URLs)
   - Width
   - Height
   - Background Color

HTML Template

```
<HTML>
<BODY>
...

<?php
include("fetch_fpx_image.php");

$fsi_url = "fsi/fsi.swf?FPXSrc=image.fpx&FPXTilesx=4&FPXTilesY=3";
$width = 320;
$height = 300;
$bgcolor = "#FFFFFF";

$template="
<Object classid=\"clsid:D27CDB6E-AE6D-11cf-96B8-444553540000\"  codebase=\"http://
download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,65,0\"
width=\"$width\" height=\"$height\">
       <PARAM name=\"movie\" VALUE=\"$fsi_url\" />
       <PARAM name=\"bgcolor\" VALUE=\"$bgcolor\" />
       <PARAM name=\"menu\" VALUE=\"false\" />
       <PARAM name=\"allowscriptaccess\" value=\"always\" />
       <PARAM name=\"allowfullscreen\" value=\"true\" />
       <PARAM name=\"quality\" value=\"high\" />
<EMBED TYPE=\"application/x-shockwave-flash\"
PLUGINSPAGE=\"http://www.macromedia.com/shockwave/download/index.cgi?P1_ Prod_
Version=ShockwaveFlash\" SRC=\"$fsi_url\" WIDTH=\"$width\" HEIGHT=\"$height\"
BGCOLOR=\"$bgcolor\" MENU=\"false\" ALLOWSCRIPTACCESS="always" ALLOWFULLSCREEN="true"
QUALITY="high"></EMBED>
</Object>";

echo $template;
?>

...
</BODY>
</HTML>
```

NeptuneLabs FSI Viewer

Plug-ins

FSI Plug-ins dynamically extend the functionality of FSI Viewer and can be included at runtime. Plug-ins have been introduced with FSI Viewer version 2.0.

Depending on the plug-in additional buttons might be added to the menu bar.
The sequence of these buttons results from the sequence of the plug-in definitions inside the <Plugins> section of the configuration file.

Using FSI Viewer Plug-ins

To include a plug-in you simply list the desired plug-ins in the <Plugins> section of any *.fsi configuration file (or the _default.fsi file).
The required plug-in files are located in the "/plugins" folder of the FSI setup directory.

Using FSI Plug-ins

```
<fsi_parameter>
...
<Plugins>
      <Plugin src="mousemodes" />
      <Plugin src="history" length="10" />
</Plugins>
...
<fsi_parameter>
```

The "Src" attribute which applies to all plug-ins specifies the plug-in to be included:

Src	
Description	Name of the plug-in
Syntax	String
Default	—
Context	<Plugins>
Version	2.0.0 or higher

This attribute defines the name of the plug-in, e.g. "history". The value has to be stated without the file extension ".plg".

Alternatively you can integrate plug-ins by HTTP query using the "plugins" parameter with a comma separated list of the desired plug-ins.

Using the Src Parameter

```
<Plugin src="mousemodes" MenuOffset="10" />
```

This option is available from FSI version 3.0 only.

Using the Src Parameter

```
<PARAM name="movie" VALUE="/fsi/fsi.swf?plugins=mousemodes,history">
```

The example above integrates the plug-in "mousemodes" and the plug-in "history".

Deactivating Plug-ins

To disable previously integrated plug-ins (e.g. plug-ins defined in the _default.fsi file) you can set a parameter with the name of the plug-in and a value of "false" to the <Options> section of your configuration file or provide a corresponding parameter via HTTP query.

Disabling the History Plug-in

```
<Options>
...
       <History value="false" />
...
</Options>
```

or by HTTP query:

Using a Query to disable History Plug-in

```
<Object
...
       <PARAM NAME="movie" VALUE="fsi.swf?cfg=image.fsi&history=false">
...
</Object>
```

Defining Plug-in Parameters

Plug-in parameters are usually defined as attributes of the <Plugin> tag.

Please note that some plug-ins might require specific attributes or even entire XML sections in the configuration file to work properly. For information on a specific plug-in please refer to the *Plug-in Reference*.

Defining plug-in parameters

```
<Plugins>
...
       <Plugin src="navextension" MenuOffset="10" visible="true" />
       <Plugin src="zoommeter">
              <SRCRelative value="true"/>
       </Plugin>
...
</Plugins>
```

Defining Plug-in Parameters dynamically

To provide plug-in parameters by query you have to prefix the corresponding parameter with the name of the plug-in and an underscore character:

PluginName_ParameterName=value

Providing plug-in parameters via HTTP query

```
<fsi.swf?navextension_visible=true&history_length=10
```

The example above sets the "visible" parameter of the "NavExtension" plug-in to "true" and the "length" parameter of the "History" plug-in to "10".

Similar to providing plug-in parameters you can provide plug-in parameters in the options section of FSI Viewer configuration files by prefixing the parameter name with the name of the plug-in. This way you can specify plug-in parameters without actually including the plug-in again.

Providing plug-in parameters as global parameters

```
<Options>
...
       <history_length value="10" />
...
<Options>
```

NeptuneLabs FSI Viewer

FSI Pages Add-on

Using FSI Pages you can present image collections on-line simulating a printed catalog, booklet or photo album.
FSI Pages automatically creates a page index with thumbnails and provides interactive zooming using FSI Viewer.

The number of pages (images) has a very low impact on the loading time and performance. You can therefore use FSI Pages with 1,000 and more images or pages. Depending on the configuration, FSI Pages can be used to display different types of booklets like catalogs (with or without links), booklets, photo albums and more.

Using FSI Pages

The integration of FSI Pages is very similar to an FSI plug-in. To enable FSI Pages you have to add a plug-in node to your XML configuration file:

Integrating FSI Pages

```
<Plugins>
...
       <Plugin src="pages" skin="silver" />
...
</Plugins>
```

Parameters for FSI Pages can be provided by adding parameters as attributes or child nodes of the FSI Pages plug-in node. As with all FSI plug-ins, you can additionally provide parameters by adding the parameter to the FSI Viewer query prefixed by "pages_".

Providing parameters as node attributes

```
<Plugins>
...
       <Plugin src="pages" MarginLeft=»100» zoom=»false» />
...
</Plugins>
```

Providing parameters as child nodes

```
...
<Plugins>
...
       <Plugin src="pages" MarginLeft="100" zoom="false" >
              <MarginLeft value=»100» />
              <zoom value=»false» />
       </Plugin>
...
</Plugins>
...
```

Providing parameters using HTTP queries attached to the movie URL

```
fsi.swf?cfg=foo.fsi&pages_marginleft=100&pages_zoom=true
```

FSI Pages Converter

FSI Pages Converter is a Java based conversion tool to split PDF documents into separate TIFF files optimized for use with FSI and eRez imaging server. Besides converting PDF documents into individual TIFF files, the conversion tool automatically extracts hyperlinks from the PDF document and generates partial PDF documents of the document for download.
Please contact your FSI dealer to get a copy of FSI Pages Converter.

Aspect Ratio of the Pages

By default FSI Pages uses the aspect ratio of the first image in the image collection to determine the aspect ratio for all pages. Alternatively you can manually define an aspect ratio for all pages using the "ratio" parameter.

E.g. `<Plugin src="pages" ratio="1:2" />`
or `<Plugin src="pages" ratio="640:480" />`

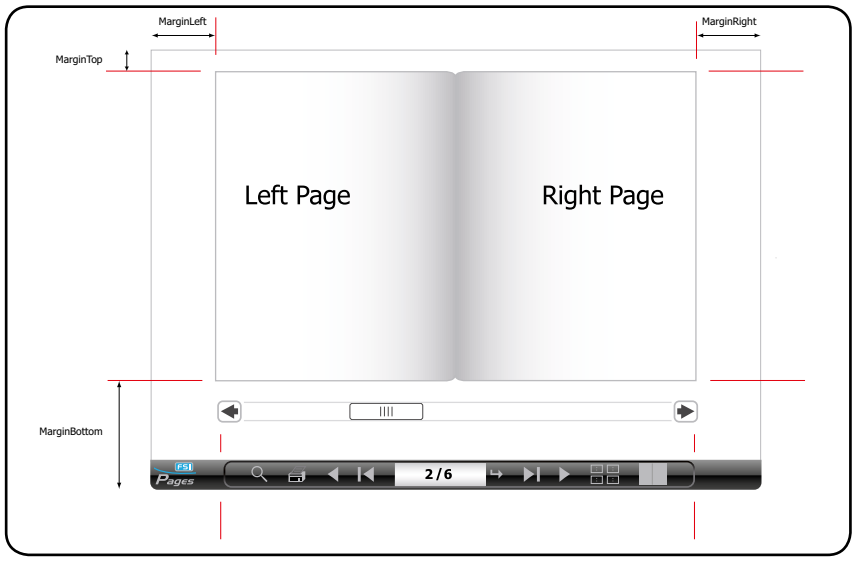
Layout and Skins

FSI Pages ships with two default skins that correspond to the default skins of FSI Viewer and an additional skin providing the user interface in a side bar.
You can easily change the FSI Pages skin being used by specifying the file name of the skin using the FSI Pages "skin" parameter e.g.

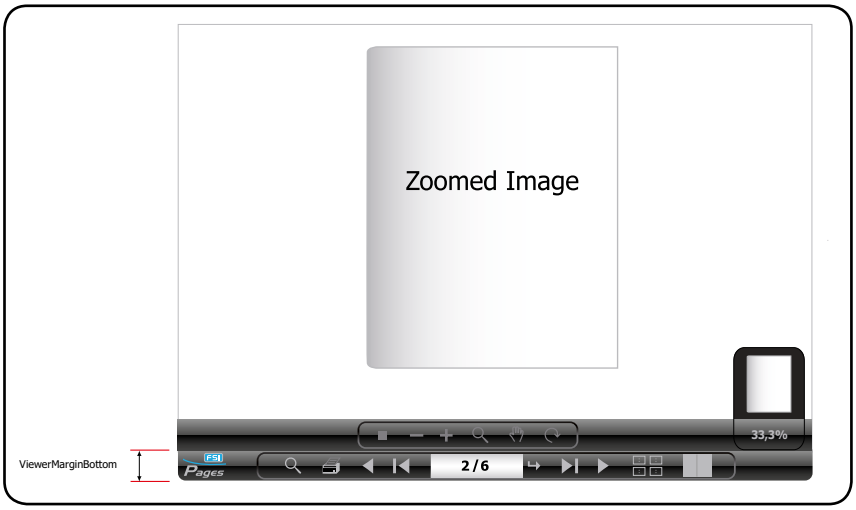
<Plugin src="pages" skin="silverbar" />

Additional custom skins are available on demand and may contain additional functionality and designs.

You can modify the layout of FSI Pages and FSI Viewer using the "Margin" and "ViewerMargin" parameters. The following illustrations show the use of these parameters:



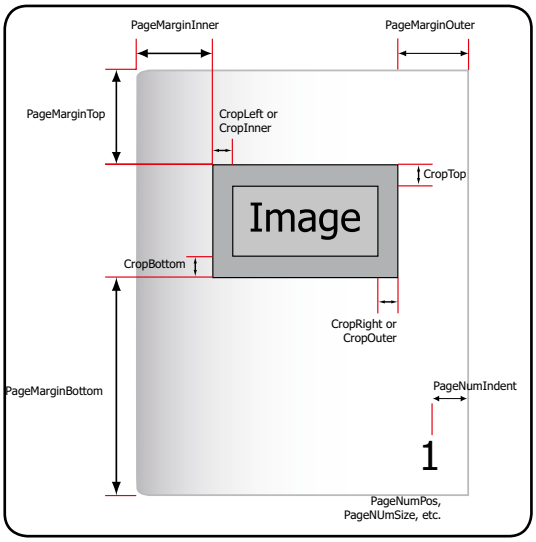
Using Margin parameters to define the position of the pages and to provide space
for the FSI Pages user interface.



Using ViewerMargin parameters to define the size and position of FSI Viewer inside FSI Pages.
Usually you only specify ViewerMarginBottom to provide space for the FSI Pages interface.

Additional layout parameters provide the possibility to adjust the layout of the individual pages. For catalogs the layout is usually very simple as the images cover the entire page. For other types of booklets like photo albums you can modify margins, alignments and page numbers and add overlays for all pages. The following illustration provides an overview on page layout parameters:
The "Crop" parameters are especially useful to remove crop marks from source images derived from print catalogs. The left and right crop margins can be defined directly or alternating depending on even/odd pages using "CropInner" and "CropOuter" parameters. The crop amount can be specified in percent or in pixels.

A complete reference of all *layout parameters* can be found at the end of this chapter.



Defining Image Collections

The images to be displayed in FSI Pages have to be defined in an <images> node added to your XML configuration file. The format and the options of the collections are the same as those of the *FSI Showcase Add-on* collections.

Defining FSI Pages image nodes

```
<Images>
...
       <Image file="image1.fsi" label="My First Image" />
       <Image file="image2.fsi" label="My Second Image"/>
...
</Images>
```

There are three different ways of defining image lists:

1. Retrieving automatically generated image lists from FSI Server
   The most convenient way to create image collections is using the template system of FSI Server. You can either do so by using the "Publish to Web" option in FSI Server web interface or by specifying the server address and a directory or search query manually.

Retrieving Directory based Image Collections from FSI Server

```
<Plugins>
...
       <Plugin src="pages" server="..." dir="..." />
...
</Plugins>
```

Retrieving Search based Image Collections from eRez Imaging Server

```
<Plugins>
...
       <Plugin src="pages" server="..." query="..." />
...
</Plugins>
```

This way FSI Server uses an image *list template* to return the matching images in XML format. Depending on the list template, FSI Server serves IPTC data belonging to an image (e.g. for links) as well.

Please note that you can only specify a directory or search query in plain text if you are using eRez and the image share you access allows public access (list role) to the images. Otherwise you need to use an encrypted string representing the list provided by eRez in the "Publish to Web" section of the web interface.

Retrieving Directory based Image Collections from eRez Imaging Server

```
<Plugins>
...
       <!– Passing a directory as plain text -->
       <Plugin src="pages" server="..." dir="images/flowers/" />

       <!– Passing an encrypted string representing a directory -->
       <Plugin src="pages" server="..." dir="*XzPzLUFVSHtmeeFxOY0NUU0" />
...
</Plugins>
```

2. Creating image nodes referencing external XML configuration files

Creating collections manually using XML configuration files

```
<Images>
...
	<Image file="image1.fsi" label="My First Image" />
	<Image file="image2.fsi" label="My Second Image" />
...
</Images>
```

3. Creating image nodes containing complete image configurations

Creating collections manually with complete XML configurations

```
<Images>
...
	<Image label="My First Image">
		<FPX>
			<SRC value="image1.fpx" />
		</FPX>
	</Image>
	<Image label="My Second Image">
		<FPX>
			<SRC value="image2.fpx" />
			<Width value="8096" />
			<Height value="12300" />
		</FPX>
		<Options>
			<NoNav value="true" />
		</Options>
	</Image>
...
</Images>
```

Method 2 and Method 3 can be freely combined while Method 1 retrieves a complete image list ignoring previously defined <images> nodes.

Image Order
The order of the images (this is the page order) corresponds to the order of the <image> declarations within the <images> section.

Inserting Blank Pages
You can insert blank pages by adding <image empty="true" /> nodes anywhere in your image collection. Alternatively you can use the FSI Pages parameter "*EmptyImages*" to pass the page numbers to insert blank pages before seperated by commas.
This is especially useful if you retrieve the image collection from FSI Server (Method 1).

Removing certain Images from a List
Using the parameter "RemovePages" you can disable (remove) certain images from an image list. In this case you pass the page number(s) of the pages you want to remove separated by commas.
This is especially useful if you retrieve the image collection from FSI Server (Method 1).

Page Reading Order

(requires FSI Pages version 4.1.5 or above)

By default FSI Pages displays pages in western reading order: left-to-right.
For documents with right-to-left reading order (e.g. Arab documents) you can change the reading order to RTL (right-to-left) using the boolean FSI Pages parameter "PageOrderRTL".

Defning Right-To-Left reading order

```
<Plugins>
...
       <Plugin src="pages" >
              <PageOrderRTL value="true" />
       </Plugin>
...
</Plugins>
```

Changing the page order to RTL changes all aspects of FSI Pages related to the page sequence and page numbering including the page sequence, the page numbering and the thumbnail display.

Please note that the page adressing does not change so that e.g. setting the "IntitialPage" parameter to "10" shows the contents on startup independet of the "PageOrderRTL" parameter. The same applies to "#page=*n*" hyperlinks in PDF documents.

If you just want to reverse the page sequence without changing the reading order, you can use the booelan FSI Pages parameter "ReversePageOrder".
Printing Pages

If you want to enable users to print pages from within FSI Pages you can enable the built-in print function by setting the FSI Pages parameter "print" to "true".

Enabling Print Options

```
<Plugins>
...
       <Plugin src="pages" >
              <Print value="true" />
              <PrintResolution value="1024" />
       </Plugin>
...
</Plugins>
```

This adds a "Print" button to the menu bar. Clicking the "Print" button displays the "Print" dialog of FSI Pages. The "Print" dialog offers different printing options to the user, depending on the current page and if the user zoomed into a page.
The printing options cover:

- Both Pages on a single sheet of paper
- Left Page
- Right Page
- The currently zoomed image section

After the user has chosen an option and clicks the "Print" button in the dialog, FSI Pages downloads the required image data (pixel data) from the imaging server and opens the print dialog on the user`s system once the download has finished.
You can limit the resolution used for printing using the FSI Pages parameter "*PrintResolution*". Lower resolution leads to faster downloads at the cost of decreased print quality.

Depending on the image server accessed you can additionally pass effect parameters "*PrintEffects*" to sharpen the image or to specify the image compression. With eRez server the corresponding parameter value to sharpen the image by

"200" and set compression to "90%" is: `sharpen=200&quality=90`

E.g. with eRez imaging server you might additionally want to use a specific real time template used to retrieve the image for printing. Please use the FSI Pages parameter "*PrintTemplate*" in this case.

Saving Pages

If you want to enable users to download a file related to the displayed page, the current section or the entire document, you can enable the save options by setting the FSI Pages parameter "save" to "true".

Enabling Save Options

```
<Plugins>
...
       <Plugin src="pages" >
              <Save value="true" />
              <SaveAllowSavingImages value="true" />
       </Plugin>
...
</Plugins>
```

Without any additional preparation this allows users to download an image of the left or right page.

Saving PDF or other document types
In addition to downloading images you might want to enable users to download the complete source PDF or specific parts of the source PDF documents. This documentation refers to PDF documents only, but you can also provide other file types for download.

1. **Downloading the entire PDF Document**
   Enabling this option is as easy as adding the FSI Pages parameter "SaveDocumentFile" providing the absolute URL to a PDF document to download.

   Example:
   `<SaveDocumentFile value="http://foo.com/somepdf.pdf" />`

2. **Downloading single or double pages**
   In order to enable the user to download a PDF document containing the page currently viewed in FSI Pages, the XML based data for a page needs to contain the corresponding URL or file name to that document.

   If you are using FSI Pages Converter to convert your PDF document, you only need to enable the "PDF Output" option in FSI Pages Converter and FSI Pages Converter splits your source document into multiple, partial PDF documents.

   You will then need to upload these files to some web server to provide them for download and add the FSI Pages parameter "`SaveURLPrefix`" specifying the URL to the directory on your server, containing these files.

   Example:
   You uploaded the PDF documents created by FSI Pages converter to your server and the URL to download the files is e.g.
   http://www.foo.com/downloads/catalog1/*.pdf

For this example your configuration should look like this:

Enabling Save Options

```
<Plugins>
...
   <Plugin src="pages" >

       <Save value="true" />
       <SaveDocumentFile value=" http://www.foo.com/downloads/catalog1/all.pdf" />
       <SaveURLPrefix value=" http://www.foo.com/downloads/catalog1/" />

   </Plugin>
...
</Plugins>
```

Note for users of imaging servers other than FSI or eRez Imaging Server
If you are using an imaging server other than FSI or eRez server you need to enable the "XML Output" option in FSI Pages Converter before converting a PDF document to have FSI Pages Converter write the XML data for each page to separate files rather than adding the data to the IPTC data of each TIFF file.
Please refer to the chapter "*Using external page data files*" for detailed instructions.

3. **Downloading parts of the Document**
   If your document is separated into logical parts (e.g. chapters or product groups), you might want to enable the user to download parts of the document. In this case you need to use the FSI Viewer plug-in "*Chapters*" which enables the user to select and go to a predefined section in your document. For each section you can specify a separate document to offer for download. This way the user will be presented an option to download the current section (chapter) in the "Save" dialog.

Example: XML data for the Chapters plug-in including download files

```
<Indexdata>
...
       <Index page="1"  label="Chapter 1" file=»chapter1.pdf» />
       <Index page=»20» label=»Chapter 2» file=»chapter2.pdf» />
       <Index page=»30» label=»Chapter 3» file=»chapter3.pdf» />
...
</Indexdata>
```

Please note:
If you use the FSI Pages parameters "SaveUrlPrefix" and/or "SaveUrlSuffix" all "file" attribute values will be prefixed and/or suffixed with the values you specified. This way you can enter the file names only and provide the path to the files using the FSI Pages parameter "SaveUrlPrefix".

Searching in FSI Pages

FSI Pages provides an optional search dialog to perform full text search in documents. For performance reasons FSI Pages does not perform the actual search itself. Instead it accesses FSI Server or any other server capable of returning search results in XML format.

To enable the search dialog in FSI Pages you simply need to set the FSI Pages parameter "Search" to true. This will add a button to FSI Pages to show or hide the search dialog. Depending on the dimension of the FSI Pages instance you might want to increase the size of the search dialog using the FSI Pages parameters "SearchDialogWidth" and "SearchDialogHeight".

By default FSI Pages adds wildcards "*" to the search keywords the user entered. You can disable this feature in case you prefer displaying exact matches or if you are using an external server to return search results that performs wildcard searching by default.

Using FSI Server to return Search Results
By default FSI Pages uses FSI Server to retrieve search results. FSI Pages passes the term entered by the user to FSI Server and receives an XML based list of matches. Based on these matches FSI Pages presents a list of thumbnails of those pages matching the search criteria.

How does FSI Server determine matches?
By default FSI Server searches the IPTC data sets that FSI Pages Converter stores the keywords in only: iptc.caption and iptc.fsi_search_data.
If your images do not yet contain keywords you might want to add keywords to the corresponding IPTC data sets of your images. If you are using a PDF document as a source for your FSI Pages instance you can have FSI Pages Converter version 2 create IPTC keyword data for all pages based on the content of the PDF document. Please refer to the "Text Search" section of FSI Pages Converter in this case.

In case you want FSI Pages to search all (or other) IPTC data sets, please use the FSI Pages parameter „SearchQueryTemplate". The default value when using FSI Server is: iptc.caption:`%%searchtext%%%20iptc.fsi_search_data:%%searchtext%%`
which limits the search to the two given IPTC sets. To search in all data sets available, modify the parameter value to: `%%searchtext%%`

Using external Severs to return Search Results
Alternatively to using imaging server you might want to use an existing database on an external server to perform the search. In this case FSI Pages provides a simple interface you can use to have FSI Pages direct the search to and retrieve the results from the external server. This is by nature an advanced feature and requires some server side scripting and database knowledge. The following section describes the required steps based on the assumption that your database contains a unique ID for each FSI Pages instance (catalog) and the required keywords.

Directing the search query to your server side script
Please use the FSI Pages parameter "SearchCustomURL" to provide the URL to the script on your server performing the search and returning the XML based results. To identify the FSI Pages instance (catalog) to return results for, you might need to pass a value identifying the catalog. This can be done using the FSI Pages Parameter "SearchCustomValue". Both parameters can – as usual – be specified by query or by adding the parameter to a configuration file.

Configuration parameters for external searching

```
<Plugins>
...
       <Plugin src="pages" >
              <SearchCustomURL value="http://foo.com/search.php" />
              <SearchCustomValue value="12345" />
       </Plugin>
...
</Plugins>
```

Using the parameters like in the example above will force FSI Pages to direct the search to the URL specified using the HTTP POST method. The POST data contains the following values:

searchvalue	The search text entered by the user
customvalue	The value you specified using "SearchCustomValue"

You might want to pass multiple, comma separated values to "SearchCustomValue" in case you need multiple values to identify a given FSI Pages instance.

Returning the XML based list of matches to FSI Pages
Your script needs to return the search result in XML format. Each result must be returned as a child node of the root node <result> (see example below). Each result item must at least contain the page number corresponding to the match. Additionally each result item may contain HTML formatted text describing the match. The sequence of the result item nodes specifies the sequence of the results displayed in FSI Pages search dialog. You therefore might want to sort the items by relevance before writing the matches to the list. Please use utf-8 encoding in case the result contains language specific, special characters.

Search Result (XML)

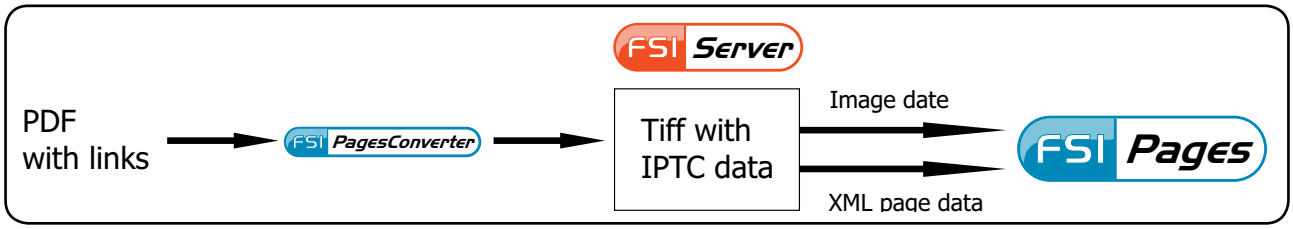
```
<?xml version="1.0" encoding="utf-8"?>
<Results>
	<Item page="10">
		Some <b>text</b> for this result...
	</Item>
	<Item page="20">
		Some <b>text</b> for this result...
	</Item>
	<Item page="30">
		Some <b>text</b> for this result...
	</Item>
</Results>
```

Adding Hyperlinks to Pages

You can add interactive areas (links) on pages by providing XML data for each page specifying the coordinates, the tool tip and the action to execute.
You might for example want to publish a PDF document containing links. In this case you can use FSI Pages Converter to extract the links from the PDF document and create the required XML data for FSI Pages for you.

Publishing PDF Documents with Links using FSI Pages Converter and FSI Server

The easiest way to add links to FSI Pages is using a source PDF document containing hyperlinks. This way FSI Pages Converter prepares the required XML data by gathering the link information from the PDF document.
FSI Server subsequently serves the XML data to FSI Pages.



1. Convert the source PDF document using FSI Pages Converter.
   Make sure to check the «*Include Links*" option in the "Hyperlinks" option section. This way FSI Pages Converter extracts link information from the PDF document.
   Make sure to uncheck the "*Enable XML Output* " option in the "*XML Output*" option section. This way FSI Pages Converter writes the XML data directly to the IPTC data of the converted TIFF files.

2. Upload the TIFF files (containing the IPTC data) to a directory on FSI Server

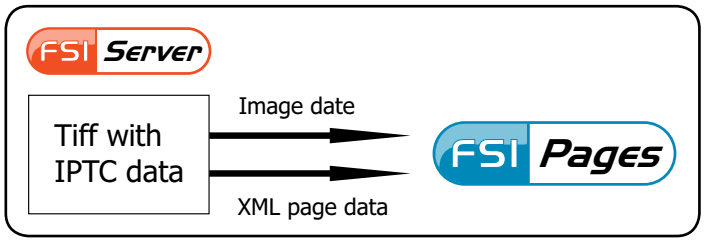
3. Go to FSI Server Interface and select the directory containing the TIFF files

4. Select "*Publish to Web*", "*Publish as FSI Pages"* Make sure to use a configuration preset with links, e.g. "Catalog with links".

Providing XML page data to FSI Pages

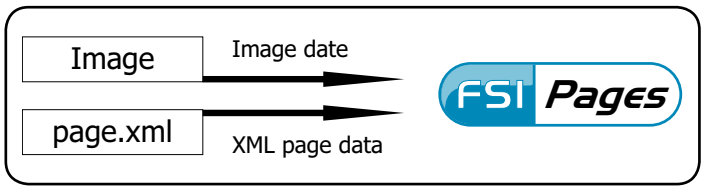
There are three alternative ways to provide XML data like links for each page.

a) Using FSI Server and IPTC data fields



FSI Server is capable of serving the IPTC data contained in an image file to FSI Viewer (FSI Pages). By default FSI Pages Converter adds the required IPTC data to the output TIFF files. The IPTC data for each page can then automatically be retrieved by FSI Pages using the FSI Server template system. This is the way it works when using "Publish as FSI Pages with links" in FSI Server web interface. A great advantage of this option is, that the data persists even when renaming or moving images.
For imaging servers not capable of serving IPTC data to FSI Pages, you need to choose one of the alternative methods below.

b) Using individual XML files



An alternative way to provide XML page data is using external XML files.
With this method each image requires a related XML file containing the page data stored in a directory on your web server. The XML files have to be named after the related images with the additional suffix ".xml".

Example:
The XML file name for the image "page065.tif" must be "page065.tif.xml".

Usually FSI Pages Converter creates the required files for you when converting a PDF document. Please enable the option "XML Output" of FSI Pages Converter in this case.

Example:
You converted a PDF document "catalog1.pdf" using FSI Pages Converter and enabled the "XML Output" option. In the output directories you specified in FSI Pages converter you can now find the TIFF files and the related XML files, e.g.

page0001.tif page0001.tif.xml
page0002.tif page0002.tif.xml
...

Now you need to upload the images to your imaging server and upload the XML files to a directory on your web server.

Finally you need to provide FSI Pages with the information where the XML files can be found on your web server. Please add the Parameter "**`XMLBase`**" to your configuration for this purpose.

Say the XML files can be accessed via:

http://www.myserver.com/xmldata/page01.tif.xml
http://www.myserver.com/xmldata/page02.tif.xml
...

The corresponding parameter for FSI pages looks as follows:

XML Page data provided by XML files

```
<Plugins>
...
       <Plugin src="pages" >
              <XMLBase value="http://www.myserver.com/xmldata/" />
       </Plugin>
...
</Plugins>
```

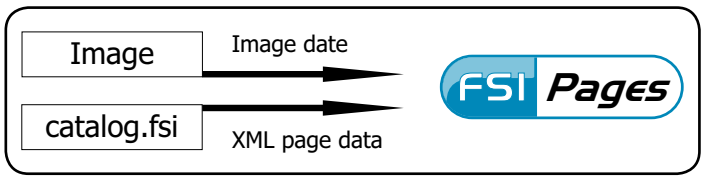
This way FSI Pages reads an XML file for each page from the location specified.

Please note that the XML files have to reside in the same domain as FSI Viewer.

c) Using <pages> nodes in configuration files

The third method is adding a <pages> node to the configuration file used for a page. Note that this method requires an individual configuration file for each page.

The example below shows the structure of page data added to a configuration file:



XML Page data in configuration files

```
<fsi_parameter>

<FPX>
 ...
</FPX>

<Pages>
       <Page file="page065.tif.xml">
              <Links>...</Links>
       </Page>
</Pages>

</fsi_parameter>
```

XML Data Format for Pages

Usually you don't need to edit the XML data for FSI Pages manually, as FSI Pages Converter creates the XML data for you. The description below provides information on the format in case you want to edit the data manually or if you want to use a different tool to create the page data.

XML Page data with 2 links

```
<Page file="page16.pdf">
	<Links urlprefix="http://somesite.com/product.php?id=">
		<Area tip=»Some Tip» url=»599»>
			<Shape coords=»0.20,0.73, 0.25,0.73, 0.25,0.72, 0.2,0.72» />
			<Shape coords=»0.35,0.73, 0.41,0.73, 0.41,0.72, 0.35,0.72» />
			<Text>Some long descriptive text</Text>
		</Area>
		<Area tip=»Another Tip» url=»210»>
			<Shape coords=»0.20,0.73, 0.25,0.73, 0.25,0.72, 0.2,0.72» />
		</Area>
	</Links>
</Page>
```

<page> node
This is the root node for XML page data. The following attributes will be recognized by FSI Pages:

Attribute	Description
File	URL or file name to use for the *save functionality of FSI Pages*

<page> node

```
<page file=»page16.pdf» />
```

<links> nodes
These nodes encapsulate one or more <area> nodes. Each <area> node represents a link. You can use the following attributes to specify default values for all child <area> nodes. Using multiple <links> nodes you can use different default values for the <area> nodes contained.

Attribute	Description
URLPrefix*	Prefix to add to all URLs specified in the child <area> nodes
URLSuffix*	Suffix to add to all URLs specified in the child <area> nodes
DefaultURL	Default URL to use for <area> nodes without a "URL" attribute
DefaultTarget	Default target frame for <area> nodes without a "target" attribute
DefaultTip	Default tool tip for <area> nodes without a "tip" attribute

* The prefix and suffix will be used for relative links only.
Special links beginning with "javascript:", "mailto:" or "#page=" will be skipped as well.

<links> node

```
<Page>
	<Links URLPrefix="http://foo.com/" DefaultTip="No Top">
		<area>...</area>
	</Links>
</Page>
```

<area> nodes
This node represents a link on a page.
Each link must contain at least one <shape> node describing the link area.
The table below lists the attributes recognized by FSI Pages.

Attribute	Description
URL	URL to open when the user clicks the link.
Target	Target frame to open the URL in
Tip	Tool tip to display when the user points at the link

<area> node

```
<Page>
	<Links>
		<Area URL="info.html" URLTarget="_blank" Tip="Hello" >
			...
		</Area>
	</Links>
</Page>
```

<shape> nodes
These child nodes of the <area> node provide the coordinates of a link polygon.
The format for the coordinates is "x1,y1, x2,y2, …".
The coordinates are floating point values between 0 and 1, where 0 is the left side of the image and 1 is the right side of the image. For y-coordinates 0 is top of the image and 1 is the bottom o the image.
A valid shape must at least consist of 6 values (3 points).

You can use multiple <shape> nodes in an <area> node if you want to specify multiple shapes for the same link.

Attribute	Description
Coords	Comma separated coordinates describing the link shape (see above)

<shape> node

```
<Page>
	<Links>
		<Area>
			<Shape coords="0,0, 1,0, 1,1, 0,1, 0,0" />
		</Area>
	</Links>
</Page>
```

The example describes a rectangular shape filling the entire image.

Please provide larger shapes first when providing overlapping shapes. FSI Pages stacks shapes above each other in the order they have been defined.

You can use any closed polygon shape. Please note that complex polygons might lead to reduced page flip performance. FSI Pages will automatically close non-closed polygons by connecting the last point to the start point.

<text> nodes (optional)
Each <area> may contain a <text> node providing HTML formatted text providing a description for the link. The *LargeToolTips plug-in* uses the content of this node to display tool tips for hyperlinks.

<shape> node

```
<Page>
	<Links>
		<Area>
			<Shape coords="0,0, 1,0, 1,1, 0,1" />
			<Text><b>Some descriptive text</b></Text>
		</Area>
	</Links>
</Page>
```

Link Parameters and Hierarchy of Link Parameters

FSI Pages provides a number of ways to define link properties.
This provides a flexible way to configure links without the need to change the page XML data, but adds a certain level of complexity.

This chapter describes how FSI Pages assembles link URLs and the hierarchy of link parameters in general.

Example: Hierarchy of "LinkURL" parameters and attributes

Parameter	XML Attribute	Location	Description
ForceLinkUrl		FSI Pages parameter	Force this URL for ALL links
	URL	**<area>** node	URL for this link
	DefaultURL	**<links>** node	Default URL for all child links
DefaultLinkUrl		FSI Pages parameter	Default URL for links

FSI Pages checks if there is a global FSI Pages Parameter "ForceLinkURL" first.
In this case all link URLs will be overwritten with this parameter.

If the <area> node does not contain an URL attribute, FSI Pages uses the DefaultURL attribute of the parent <links> node. If this attribute is undefined, FSI Pages uses the global FSI Pages Parameter "DefaultLinkURL".

The same process applies to the "target", "tip", "prefix", "suffix" and "javascripttarget" parameters:

Force-Parameter	Default-Parameter	<links> node	<area> node
ForceLinkURL	DefaultLinkURL	DefaultURL	URL
ForceLinkTarget	DefaultLinkTarget	DefaultTarget	Target
ForceJavascriptTarget	DefaultJavascriptTarget	DefaultJavascriptTarget	Target
ForceLinkTip	DefaultLinkTip	DefaultTip	Tip or <text> child node
ForceLinkURLPrefix	DefaultLinkURLPrefix	URLPrefix	n/a
ForceLinkURLSuffix	DefaultLinkURLSuffix	URLSuffix	n/a

Relative and Absolute Links

FSI Pages does not add prefixes or suffixes to absolute link URLs. In this case any prefixes and suffixes will be ignored.

Absolute Link URL

```
<Page>
	<Links>
		<Area URL="http://www.fsi-viewer.com">...</Area>
	</Links>
</Page>
```

Resulting URL: *http://www.fsi-viewer.com*

The same applies to *special URLs* like "*javascript:*" and "*mailto:*" as well as *FSI Pages command URLs* beginning with "#".
For relative links FSI Pages adds the URLPrefix and/or URLSuffix values if specified.

Relative Link URL

```
<Page>
	<Links URLPrefix="http://foo.com/" URLSuffix="&test=1">
		<Area URL="products.html?id=123">...</Area>
	</Links>
</Page>
```

Resulting URL: *http://www.foo.com/products.html?id=123&test=1*

Special URL Values

For some special URL values FSI Pages ignores the URLPrefixes and URLSuffixes as well. The following table lists these special URLs:

URL	Description
javascript:*foo()*	Call the JavaScript function *foo()*
mailto:email@domain.tld	Send an email to email@domain.tld
#none	Do nothing. Handy if you just want to display a tool tip. **Available in FSI Pages version 4.0.7 or above**
#page=*n*	Forces FSI Pages to flip to page *n*
#showimage=[imageURL]	Displays the specified JPEG image
#zoom	Forces FSI Pages to zoom to the area covered by the link's shape(s)
#zoomarea	Magnifies the area covered by the link's shape(s) in place
#zoomimage	Display a custom JPEG image. The value [URL] needs to be an absolute URL to a JPEG image.
#zoom=1,1,0,0,0.5,0.5	Forces FSI Pages to zoom to the image area following "=" See Parameter **>**"*InitialView*" for format details)
#zoomonpage=*n*;*strView*	Combination of "#page=" and "#zoom=". This is: first flip to page *n*, then zoom to area *strView*. **Available in FSI Pages version 4.0.7 or above**

Please note that the special links starting with "#" need to be provided exactly as listed in the table above. Using e.g. "#zoomToPage=[…]" or "foo.html#zoom" will link to the HTML anchor provided rather than executing an FSI Pages action.

For "**mailto:**" URLs the target frame is always "_self", except the XML <area> node specifies a different target.

Modifying Link Values at Runtime

In some cases it might be necessary to replace parts of link URL values for links on the pages – for example when using session variables or if you use the same catalog in HTTP and HTTPS environments. Runtime link template replacement applies to links on pages as well as for "SaveURL" values used to specify external documents related to a page.

You can replace any part of these URLs using the FSI Pages parameters "LinkTemplates" and "LinkTemplateData".

Use the FSI Pages parameter "LinkTemplateData" to specify a comma separated list of values you want FSI Pages to replace the keywords with.
Usually you will provide the LinkTemplateData value by appending the parameter to the FSI Viewer query using server side script.

When specifying multiple LinkTemplates, the number and sequence of the corresponding LinkTemplateData values must match to achieve correct replacement.

For example replacing the templates "[SESSION]" and "[DATE]" at runtime in all link URLs can be done like this:

Replacing parts of link URLs at runtime

```
Configuration:
...
<Plugins>
...
       <Plugin src=»pages» >
              <LinkTemplates value="[SESSION],[DATE]" />
       </Plugin>
...
</Plugins>
...
FSI URL
.../fsi.swf?LinkTemplateData=somesession,01.01.2006
```

This will replace all instances of "[SESSION]" by "somesession" and all instances of "[DATE]" by "01.01.2006" in all link URLs.

Please note that LinkTemplateData values have to be provided *url-encoded* if the values contain invalid HTML query characters like "?&/,".

Besides using the LinkTemplates parameter to provide custom templates, you can use the following predefined LinkTemplates without providing the corresponding LinkTemplateData:

Template	Description
_FSI_CURRENTPAGE_	1-based index of the current page (1,2,3,4…)
_FSI_CURRENTDOUBLEPAGE_	1-based index of the current double page (1,2,4,6…) For inner pages this value represents the index of the left page. For the front and back cover the value is the actual page index

Page Overlays

Page overlays are compiled Flash movies clips that can be used as an overlay, floating above a page in FSI Pages. You have probably seen the "Photo Album" example created with FSI Pages at *www.fsi-viewer.com*. The small sticky photo corners come from an overlay file.

Using Overlays
Overlay files have to reside in the "/plugins/pages/overlays/" directory of your FSI Viewer setup directory. You can add a global overlay to appear on all pages and/or specify an overlay file for specific pages.

Defining a global page overlay

```
<Plugins>
...
       <Plugin src="pages" >
              <PageOverlays>
              <Overlay default="true" file="ov_photo.swf" />
              [...]
              </PageOverlays>
       </Plugin>
...
</Plugins>
```

The example above adds the "sticky photo corners" overlay to all pages, like in the "Photo Album" sample.

Defining overlays for specific pages

```
<Plugins>
...
<PageOverlayShowZoomed value="true" />
<PageOverlays>
        <Overlay file="myDefaultOverlay.swf"  />

        <Overlay page="5" file="myOverlay1.swf" />

        <Overlays page="10">
             <Overlay file="myOverlay2.swf" x="0.5" y="0.5" />
             <Overlay file="myOverlay3.swf" />
        </Overlays>

</PageOverlays>
...
</Plugins>
```

The example above does the following:

- add a default overlay "myDefaultOverlay.swf"
- adds the overlay file "myOverlay1.swf" to page 5
- adds"myOverlay2.swf" and "myOverlay3.swf" to page 21

The attributes of the <overlay> nodes can be used to specify variables for an overlay.
Note that you can, but don't need to define a default overlay.

Please refer to the *FSI Contrib package* if you want to modify or create custom page overlay files.

FSI Pages Presets in FSI Server Interface "Publish to Web"

Using FSI Pages with FSI Server imaging server you can choose from the following presets when using the "Publish as FSI Pages" option:

- **Catalog with links**
  Each image represents a page, no page margins, no overlays
  If the image contains XML data in the "IPTC_FSI" data field, links will be presented on the page.
- **Hard cover Book**
  First an dlast page of the catalog have a inflexible appearance.
- **Teaser Thumbnail**
  No user interface, zoom is disabled. Pages flip automatically when idle.
  You can specify a link for the entire viewer area so that you can use this template to publish a thumbnail of a catalog linking to the full size catalog.
  See *Parameters for FSI Pages Thumbnails* for configuration details
- **Photo Album**
  A simple photo album with photo corners and an emboss effect for each image.

Event Notifications and Actions

When using JS Bridge plug-in or FSI Viewer component you can use the following event notifications and actions to interact with FSI Pages Add-on.
Please refer to JS Bridge plug-in for details on how to setup communication between JavaScript and FSI Viewer.

Event Notifications

Event ID	Parameter	Description
onPagesInitComplete	Integer	Called after the FSI Pages Add-on is ready for use. The parameter provides the total number of pages.
onPagesGotoPage	Integer	Called each time before the user goes to a different page. The parameter provides the target page number.
onPagesPageChanged	Integer	Called each time after the current page has changed. The parameter provides the index of the page currently visible.
onPagesProgress	Integer	Called each time the load progress of FSI Pages changes. The parameter provides the number of pages remaining in the load queue.
onPagesShowIndex	Boolean	Called each time the user shows (true) or hides (false) the page index.
onPagesShowZoom	Boolean	Called each time the user starts (true) or ends (false) page zoom.
onPagesLinkClick	URL	Called each time the user clicks a link. The parameter contains the URL.

Commands for FSI Pages Add-on

Command	Description
'' (empty string)	Stop current action
FirstPage	Go to first page.
LastPage	Go to last page.
NextPage	Go to next page.
PreviousPage	Go to previous page.
GotoPage	Go to page *n* **Important:** Use SetVariable("newImageIndex", *n*) where *n* is [0 .. pages] prior to sending this command Alternatively you can specify a page relative to the current page. To go to a page relatively you have to set "newImageIndex" to a number prefixed by "+" or "-", e.g. "+1" is the same as "NextPage".
Print	Open or close the print dialog.
Save	Open or close the save dialog.
Search	Open or close the search dialog.
ZoomLeftPage	Zoom the left page
ZoomRightPage	Zoom the right page
NormalPageView	Exit zoom view
ShowHidePageIndex	Show or hide the page index (thumbnail view)

Parameters to retrieve Image Collections

Server	
Description	Address of imaging server
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The URL of the setup path of imaging server to be used when retrieving image collections.

Dir	
Description	Image directory
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The directory on imaging server containing the images to use for FSI Pages.

Query	
Description	Search query
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

A search query to collect images from imaging server to be used for FSI Pages.

ListTemplate	
Description	Template used to retrieve image list from imaging server
Syntax	String
Default	"fsi/image_list.xml"
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The XML based template to use when retrieving image lists from imaging server via "dir" or "query" parameters. Available templates are located in the "WEB-INF/templates/fsi/" directory of your imaging server setup.

Alphabetical Index of FSI Pages Parameters

EmptyImages	
Description	Define blank pages
Syntax	String
Default	---
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.3.0 or higher

Using this parameter you can add blank pages. The value of the parameter must contain one or more image page numbers of the blank pages to be inserted.
E.g. `<EmptyImages value="2,10" />` adds two blank pages at page 2 and page 10, moving the other images in your collection correspondingly.
Alternatively you can add `<image empty="true" />` nodes to your image collection.

RemovePages	
Description	Remove individual pages from the collection
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Using this parameter you can remove images from the image collection. The behavior of FSI Pages is exactly as if the images would not be listed in the image collection.
E.g. `<EmptyImages value="1,3,4" />` removes the first, third and fourth image in the collection. Please note that removing images takes place before inserting blank pages using the "EmptyImages" parameter.

ReversePageOrder	
Description	Reverses the order of the individual pages in the collection
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or higher

Set this parameter to "true" to reverse the page sequence of pages. Please note that this parameter does not change the reading order. Please see *Page Reading Order* to change the reading order of a document.

Basic Parameters

InitialPage	
Description	Page to display on startup
Syntax	Number
Default	1
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specifies the page to display on startup, e.g. "1" for the front cover.

Note: The parameter "InitialPage" has no effect if you are using the "*RememberLastViewedPage"* parameter to make the user to return to the last viewed page. You can use "ForceInitialPage" instead if you want FSI Pages to show a specific page independent of the last viewed page.

PageOrderRTL	
Description	Switch to right-to-left reading order
Syntax	Boolean
Default	False
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or above

By default FSI Pages uses the western page reading order (left-to-right). Enable this parameter to use *right-to-left reading* order, e.g. for Arab documents.

FirstPageNumber	
Description	Offset for page numbering
Syntax	Number
Default	1
Context	Pages plug-in node or HTTP query prefixed with "pages_"

By default (FirstPageNumber=1) the front cover is page number one. You can specify an offset so that the page numbers start with a different value. This parameter affects page numbers, the index and the display of the current page number in the FSI Pages user interface.

Skin	
Description	User interface (skin) for FSI Pages
Syntax	String
Default	silver
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specifies the skin containing the user interface for FSI Pages. Available skins are located in the directory "skins/pages" of your FSI setup directory. Please note that this does not affect the skin of FSI Viewer which can be specified separately using the "skin" parameter of FSI Viewer.

Slider	
Description	Show / hide the page slider
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Show (true) or hide (false) the slider to select pages.

Index	
Description	Enable / disable table of contents
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Enable (true) or disable (false) the user to show the page index containing thumbnails of all pages.

KeepIndex	
Description	Keep index data when hiding the index
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Setting the value to true (default) speeds up showing and hiding the page index with a possible overall performance penalty when presenting many pages. Setting the value to false deletes index data when hiding the index, forcing the viewer to build the index each time it is shown.

PageTurnSpeed	
Description	Speed when turning pages
Syntax	Number
Default	82
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The speed of the page turning effect. "0" is the slowest and "100" the fastest value possible.

Zoom	
Description	Enable / disable zooming into pages
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Enable (true) or disable (false) zooming into pages with FSI Viewer when clicking on a page.
Zooms into the specified link area.

ShowLoadProgress	
Description	Show / hide the load progress display
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or above

Show (true) or hide (false) the load progress display. FSI Pages automatically sets this parameter to "true" if you specify a hyperlink for the entire FSI Pages instance (*see DemoURL*).

AutoCropPages	
Description	Crop images to fill entire page
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

Enable (true) this option to automatically crop images to fill the entire pages. You need to specify an aspect ratio manually in this case.

ClickPageToZoom	
Description	Enable / Disable zooming by clicking a page
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or higher

Enabling (true) this option enables the user to zoom a page by clicking it.

ZoomAreaAnimationSpeedIn	
Description	Defines the animation speed of in place maginifications initiated by "#zoomarea" hyperlinks
Syntax	Number
Default	520
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0 or above

Defines the speed of the zoom animation when zooming **in**.

ZoomAreaAnimationSpeedOut	
Description	Defines the animation speed of in place maginifications initiated by "#zoomarea" hyperlinks
Syntax	Number
Default	240
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0 or above

Defines the speed of the zoom animation when zooming **out**.

ZoomAreaAnimationType	
Description	Defines the animation type of in place maginifications initiated by "#zoomarea" hyperlinks
Syntax	Number
Default	5
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0 or above

Defines the animation type of the zoom animation.

ZoomAreaTargetScale	
Description	Magnification level of in place maginifications initiated by "#zoomarea" hyperlinks
Syntax	Number
Default	200
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0 or above

Defines the zoom factor for the link. Two limitations apply: The zoomfactor must be 110 or greater, otherwise no zoom area is displayed. The zoomarea may not exceed the FSI Pages instance size.

DoublePageAllowInteractivity	
Description	Enables the interactive zoom mode
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

When enabled the image will scroll when the user moves the mouse pointer.

DoublePageZoom	
Description	Enable or disable interactivity in double page zoom mode
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

If enabled the zoom mode will display both pages at the given zoom level.
When disabled, interactivity (like clicking hyperlinks) is disabled in double page zoom mode.

DoublePageZoomScale	
Description	Defines the zoom level of double page zoom
Syntax	Number
Default	1:1
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

Default: "1:1" (zoom to original size)
a) relative to source image resolution ("1:1", "1.5:1",?1:2?...)
b) relative to current page size in percent (200, 300, ...)

ForceXMLDataFiles	
Description	Force XML data useage
Default	false
Context	Pages plug-in
Version	4.1.5 or higher

If set to true, the IPTC data from the image file is ignored and the links and overlay information is retrieved from the XML files. If set to false, XML data is used only if no IPTC data is present in the image.

Parameters for the Front- and Backcover

FrontCover	
Description	Enable/disable the front cover
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Setting "FrontCover" parameter to false disables access to the front cover and forces FSI Pages to display the first image on the first inner left page.

BackCover	
Description	Enable/disable the back cover
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Setting "BackCover" parameter to false disables access to the back cover and forces FSI Pages to add a blank content page if required.

BlankFrontCover and BlankBackCover	
Description	Use a blank page for front and/or back cover
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Use a blank page as front and/or back cover. The cover page(s) will be added to images in the image list.

FullFrontCover and FullBackCover	
Description	Fill entire front and/or back cover with image
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Fill the entire front and/or back cover with the corresponding image. This is: Ignore page margins and do not use an overlay for the cover page specified.

FrontCoverImage and BackCoverImage	
Description	Use the specified image for front or back cover
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

A path to an image to use for the front or back cover. FSI Pages will use the "FPXBase" parameter of FSI Viewer for relative paths. The cover page(s) will be added to the images in the image list.

FrontCoverConfig and BackCoverConfig	
Description	Use the specified image configuration file for front or back cover
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specify a separate image configuration file (*.fsi) to be used for the front or back cover respectively. The cover page(s) will be added to images in the image list.

HardPages	
Description	Defines page flexibility
Syntax	Boolean
Default	false / all / cover
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

Defines wether the pages are flexible or inflexible e.g. like a book cover.

PageBackgroundImage	
Description	Relative path to the image on your imaging server
Syntax	Filename
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

The path to an image on your imaging server to be used as a background.
If you use a relative path (recommended) the FSIBase parameter will be used to get the absolute path to the image.

PageBackgroundImageEffects	
Description	Image effects for the background image
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.0

Defines image modification parameters to apply to images used. The value of this parameter depends on the imaging server being used.
With FSI Server you can sharpen the image and define the JPEG compression level, e.g. "effects=sharpen(230)&quality=95".

Parameters defining the Layout and Appearance

Ratio	
Description	Aspect ratio of pages
Syntax	String
Default	"auto"
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The aspect ratio of pages displayed in FSI Pages.
You can specify any aspect ration (e.g. "320:240" or "1:2") or specify "auto"
to use the aspect ratio of the first image in the image collection.

SingleSided	
Description	Use images on uneven or even pages only
Syntax	String
Default	false / even / uneven
Context	Pages plug-in node or HTTP query prefixed with "pages_"

By default FSI Pages creates a page for each image in the image collection.
Using this parameter you can advise FSI Pages to use images on even or uneven pages only, inserting a blank page on the opposite side.

Use "false" to display images on all pages. (default)
Use "even" to display images on even pages only.
Use "uneven" to display images on uneven pages only.

BendEffect	
Description	Visual "3D" of the pages
Syntax	"Glossy" or "Matte"
Default	"Glossy"
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

The effect "Glossy" adds highlights and shadows, while "Matte" adds shadows only (like with FSI Pages version 3).
Please use BendEffectIntensity parameter to modify the effect level or disable the visual effect.

BendEffectIntensity	
Description	Level of visual "3D" effects
Syntax	Number
Default	65
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

Defines the 3D effect level of the pages in percent. Using a value of "0" disables the visual 3D effect. The value additionally affects the opacity of the drop shadow.

DropShadow	
Description	Enable or diable the dropshadow effect for the pages
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

Enables or disables the drop shadow of the pages.

DropShadowDistance	
Description	Distance of the page's drop shadow
Syntax	Number
Default	50
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

Distance of the pages‘ drop shadow in percent, where "100" is the greatest possible distance.

ImageAlign	
Description	Alignment of images on the pages
Syntax	String
Default	"TC"
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Alignment of the images on the pages. The default is "TC" which centers the images horizontally on top of the page. Please specify the alignment for even (left) pages. Uneven (right) pages will be aligned correspondingly. Possible values are any combination of:

T (Top) C (Center) B (Bottom)	and	L (Left) C (Center) R (Right)

PageColor	
Description	Page background color
Syntax	HexColor
Default	FFFFFF
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Background color of the pages specified as 6 digit web color.

PageBorderColor	
Description	Color of page borders
Syntax	HexColor
Default	CCCCCC
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Color of the page borders specified as 6 digit web color.Use "false" for no border at all.

HelpImageURL	
Description	Help image to be displayed left of the first page
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

An absolute or relative ImageURL (like FPXSrc) to an image on the imaging server that FSI Pages displays left of the first page of the catalog.

ScrollBaseColor	
Description	Base color for the scroll bar
Syntax	HexColor
Default	CCCCCC
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

6-digit hex-color for the scroll bar in the page index.

ScrollArrow	
Description	Color for the scroll bar's arrows
Syntax	HexColor
Default	000000
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

6-digit hex-color for the arrows of the scroll bar in the page index.

ScrollTrack	
Description	Color for the scroll bar's track bar
Syntax	HexColor
Default	DDDDDD
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

6-digit hex-color for the track bar of the scroll bar in the page index.

MarginLeft, MarginTop, MarginRight, MarginBottom	
Description	Margin of the turning pages
Syntax	Number in pixel
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Margin on each side of the turning pages in pixel. The margin effects the position of the pages only, not the user interface of FSI Pages. You should at least specify a bottom margin to provide space for the interface.

PageMarginOuter, PageMarginTop, PageMarginInner, PageMarginBottom	
Description	Margin of the page content
Syntax	Number in percent of the page width / height
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Margins of the content of each page in percent of the page width (inner and outer) or page height (top and bottom).

ViewerMarginLeft, ViewerMarginTop, ViewerMarginRight, ViewerMarginBottom	
Description	Margin of FSI Viewer
Syntax	Number in pixel
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Margin on each side of FSI Viewer in pixel when zooming into a page. Usually you need to set "ViewerMarginBottom" to the height of the FSI Pages interface.

ThumbSize	
Description	Max. size of thumbnails in the page index
Syntax	Number in pixel
Default	60
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The maximum size of thumbnails displayed in the index. The actual width and height depend on the aspect ratio of the pages (see parameter "ratio").

ThumbPadding	
Description	Padding between thumbnails in the index
Syntax	Number in pixel
Default	10
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The padding between thumbnails displayed in the page index in pixel.

PageNumbers	
Description	Display page numbers on pages
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Show or hide page numbers on pages.

PageNumPos	
Description	Alignment of page numbers
Syntax	String
Default	"BL"
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Alignment of page numbers. The default is "BL" for bottom-left.
Please specify the alignment for even (left) pages. Uneven (right) pages will be aligned correspondingly. Possibly values are any combination of:

T (Top)		L (Left)
C (Center)	and	C (Center)
B (Bottom)		R (Right)

PageNumIndent	
Description	Indentation of page numbers
Syntax	Number in percent of the page width
Default	1%
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specifies the indentation in percent of the page width.
This parameter does not apply if the page number position is centered horizontally.

PageNumSize	
Description	Size of page numbers
Syntax	Number in percent of the page height
Default	8%
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Size of page numbers in percent of the page height.

PageNumColor	
Description	Color of page numbers
Syntax	HexColor
Default	000000
Context	Pages plug-in node or HTTP query prefixed with "pages_"

6-digit hex color of page numbers.

PageNumBGColor	
Description	Background color of page numbers
Syntax	HexColor
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"

6-digit hex color of page numbers or "false" for no background.

UseRomanPageNumbersToPage	
Description	Display Roman page numbers up to page number n
Syntax	Number
Default	0
Context	FSI Pages
Version	5.0.4

Defines the page number up to which FSI Pages uses Roman page Numbers.
The default value of "0" forces FSI Pages to display Roman numbers for negative numbers and 0.
If you set the value to "5" the display will be like this:
I,II,III,IV,V,6,7,8,9…

Note that you can use the Parameter "FirstPageNumber" to offset all page number values.

RomanPageNumbersOffset	
Description	Offset Roman page numbers by given value
Syntax	Number
Default	0
Context	FSI Pages
Version	5.0.4

Defines the value of the first Roman page number.
The default value of "0" means no offset so that the Roman numbers start with "I".

Parameters for Links on Pages

FollowLinks	
Description	Enable or disable FSI Pages to open links
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Setting this parameter to "false" prevents FSI Pages from opening links on the pages. Links and tool tips will nevertheless be displayed.

ShowLinksWhileFlipping	
Description	Show or hide links during flip animations
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

Set this parameter to "true" if you want FSI Pages to display link shapes during flip animations. Showing links decreases the performance of the flip animations.

LinkFillAlpha	
Description	Opacity of the link area
Syntax	Number
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

Specify the opacity of the link areas from 0 (no fill) to 100 (opaque). The color corresponds to the color of the link area`s border color.

LinkRGBANormal	
Description	Color and opacity of links in normal state
Syntax	RGBA
Default	0000FFFF
Context	Pages plug-in node or HTTP query prefixed with "pages_"

8-digit hexadecimal number specifying the color and opacity of links on the pages in normal state in the form "RRGGBBAA". E.g. "FF0000FF" for opaque red or "00FF0099" for semi-transparent green

LinkRGBAHover	
Description	Color and opacity of links in hover state
Syntax	RGBA
Default	FF00FFFF
Context	Pages plug-in node or HTTP query prefixed with "pages_"

8-digit hexadecimal number specifying the color and opacity of links on the pages when the user points at a link in the form "RRGGBBAA". E.g. "FF0000FF" for opaque red or "00FF0099" for semi-transparent green

LinkRGBAActive	
Description	Color and opacity of links in active state
Syntax	RGBA
Default	FF0000FF
Context	Pages plug-in node or HTTP query prefixed with "pages_"

8-digit hexadecimal number specifying the color and opacity of links on the pages when the user clicks a link in the form "RRGGBBAA". E.g. "FF0000FF" for opaque red or "00FF0099" for semi-transparent green

ForceLinkURL	
Description	Force URL for all links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Force FSI Pages to use this URL for all links on all pages. This overwrites any URLs specified in the page data.

ForceLinkTarget	
Description	Force a target frame for all hyperlinks on all pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Force FSI Pages to use this HTML target frame for all links on all pages.
This overwrites any target frames specified in the page XML data.
Note: this parameter does not affect "*javascript:*" and "*mailto:*" links.

ForceJavascriptTarget	
Description	Force a target frame for all "javascript:" links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.1 or above

Force FSI Pages to use this HTML target frame for all "javascript:" links on all pages. This overwrites any target frames specified in the page XML data.
Note: this parameter does not affect URL and "*mailto:*" links.

ForceLinkUrlPrefix, ForceLinkUrlSuffix	
Description	Force a link prefix / suffix for all links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Force FSI Pages to use this prefix /suffix for all links on all pages.
This overwrites any prefixes and suffixes specified in the page data.
Prefixes and suffixes will not be applied to absolute URLs and *special link values* like "javascript:".

ForceLinkTip	
Description	Force a tool tip for all links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Force FSI Pages to use this tool tip for all links on the pages.
This overwrites any tool tips specified in the page data.

DefaultLinkURL	
Description	Default URL for links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Use this URL for links if no URL has been specified in the page data.

DefaultLinkTarget	
Description	Default target frame for all hyperlinks on all pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Use this HTML target frame for links if no target frame has been specified in the page XML data.

DefaultJavascriptTarget	
Description	Default target frame for all "javascript:" hyperlinks on all pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.1 or above

Use this HTML target frame for "javascript:" links if no target frame has been specified in the page data.

DefaultLinkUrlPrefix, DefaultLinkUrlSuffix	
Description	Default prefix / suffix for link URLs on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Use this prefix or suffix for all links on the pages if no prefix / suffix has been specified in the page data.

DefaultLinkTip	
Description	Default tool tip for links on pages
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or above

Use this tool tip for all links on the pages if no tool tip has been specified in the page data.

LinkTemplates	
Description	Template identifiers for link URLs
Syntax	String(s)
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

You can provide one or more identifiers that will be replaced in Link URLs.
The identifiers are case sensitive and must be separated by commas.
The values to insert have to be defined by the parameter "LinkTemplateData" explained below.
Please refer to the chapter *Modifying link values* at runtime above for details.

LinkTemplateData	
Description	Values to replace template identifiers in link URLs
Syntax	String(s)
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

When defining "LinkTemplates" to replace place holders in Link URLs you can use this parameter to specify the values that shall be inserted. The number and sequence of the values must match the number and sequence of the templates you defined using the "LinkTemplates" parameter. Multiple values have to be separated by commas and the individual values have to be provided *urlencoded*. Please refer to the chapter *Modifying link values at runtime* above for details.

Parameters for FSI Pages Thumbnails

Due to the Single Source Imaging concept you can display a small, flipping thumbnail of your catalog just by using a different configuration file and choosing a smaller dimension for the FSI Pages instance. The parameters below focus on thumbnail pages.

Menu	
Description	Show / hide the user interface
Syntax	Bool
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Show (true) or hide (false) the user interface of FSI Pages add-on.

IdleAutoTurn	
Description	Turn pages automatically
Syntax	Boolean
Default	false
Context:	Pages plug-in node or HTTP query prefixed with "pages_"

Run the FSI Pages in demonstration mode and turn pages automatically when the user does not interact (move the mouse) for 2 seconds.

IdleAutoTurnDelay	
Description	Min. delay between automatic page flips
Syntax	Floating Point Number (seconds)
Default	2
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0

Show each page for at least n seconds when flipping pages automatically. Please note that a page might be displayed for longer if loading the page takes longer than the time specified.

IdleAutoTurnMinPage, IdleAutoTurnMaxPage	
Description	Auto flip range
Syntax	Number (page)
Default	(all pages)
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0

Loop the auto flip action between IdleAutoTurnMinPage and IdleAutoTurnMaxPage. This way you can restrict the auto flip to a range of pages. If the initial page is outside the range specified, FSI Pages flips towards the range before looping between the range specified.

DemoURL	
Description	Global link when clicking FSI Pages
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Specifies a link to open when the user clicks anywhere on the FSI Pages instance. This parameter is especially useful when using a small preview of a catalog to lead the user to the full size version of FSI Pages.
You can use URLs starting with "javascript:" to have a Javascript function called when the user clicks the viewer.

DemoURLTarget	
Description	Global link when clicking FSI Pages
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

HTML frame in which to open the URL specified by "DemoURL" parameter in.
You can use default frame names like "_self" or "_blank" or use a custom frame of your HTML frame set.

Rotation	
Description	Rotation in degrees of the thumbnail
Syntax	Number
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.0 or higher

Optional rotation of FSI Pages thumbnails.
Please note that using the "rotation" parameters with normal size FSI Pages instances is not recommended as zooming, printing, index view and alike do not correspond to the rotation.

PageInput	
Description	Show or hide Page Input field
Syntax	Boolean
Default	true
Context	Pages plug-in
Version	4.0.0 or higher

Defines wether the input form for page numbers is shown.

ButtonsFirstLastPage	
Description	Show or hide First/Last Page buttons
Syntax	Boolean
Default	true
Context	Pages plug-in
Version	4.0.0 or higher

Defines wether the navigation buttons for jumping to the first or last page are shown.

Parameters for Statistics

(Requires FSI Viewer 4.1.3 or above)

The following parameters can be used to measure the usage of FSI Pages instances by means of third party statistic tools. Using these parameters you can define a template based URL to call each time the user flips to another page. The templates in the URL will be replaced at runtime. You can specify a minimum time a page is visible before the statistics URL will be loaded.

PageLogURL	
Description	Template based URL for page view logging
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.3 or higher

Template based URL to load each time the user flips to another page.
The following templates will be replaced at runtime:

[pagenum] - The page number of the current left page
[totaltime] - Total seconds the FSI Pages instance is visible
[instanceid] - A unique number identifying the FSI Pages instance
[uniqueid] - The FSI Pages parameter value "UniqueID"

Example URL:
http://foo.com/log.php?catalog=foo&page=[pagenum]&iid=[instanceid]

You can alternatively call a Javascript function using the "javascript:" syntax. In this case you need to define a valid HTML target frame using the "PageLogTarget" parameter.

PageLogTarget	
Description	HTML Target Frame to open the PageLogURL in
Syntax	String
Default	_internal
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.3 or higher

By default FSI Pages uses the special target "_internal" which loads the URL from within the Flash plug-in. You can use this parameter to specify a different HTML target frame like "_self" or "_top".

PageLogDelay	
Description	Page View logging delay
Syntax	Number (milliseconds)
Default	2000
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.3 or higher

Specifies a minimum period of time a page must be visible before loading the PageLogURL. Use "0" to log each page access immediately. The default value of 2000 (ms) ensures, that a page has been viewed for at least 2 seconds.

Advanced Parameters

RememberLastViewedPage	
Description	Store the most recently viewed page locally
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.5 or above

If you enable this parameter FSI Pages stores the page number of the most recently viewed page on the user's computer and displays this page on start up the next time the user views the same catalogue (image collection) again.

Note: The parameter "*InitialPage*" has no effect if there is a valid value for the last viewed page. You can use "*ForceInitialPage*" if you want FSI Pages to show a specific page independent of the last viewed page

Note: When using image collections retrieved from FSI Server, FSI Pages uses the "*dir*" or "*query*" parameter to identify the image collection automatically. If you create image collections manually, you additionally need to add the parameter "*UniqueID*" so that the "last viewed page" can be assigned to this image collection.

RememberLastViewedPageExpireAfter	
Description	Expire time for "RememberLastViewedPage" parameter in seconds
Syntax:	Number (seconds)
Default	Undefined (infinite)
Context:	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.9 or above

By default the last viewed page will be restored on the next visit of a user if you enable the RememberLastViewedPage parameter. You might want to specify an expire time after that the last viewed page will be discarded. Please specify the expire time in seconds, e.g. "3600" for one hour.

AutoZoom	
Description	Initial magnification in FSI Viewer
Syntax	Number in percent of the source image or "fitWidth" or "fitHeight"
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

The initial magnification in FSI Viewer after the user selected an image to zoom. Specify the magnification in percent of the source image, e.g. "100" zooms to the physical resolution of the source image (default maximum magnification). With FSI Viewer 4.1 or above you can additionally use the parameter values "fitWidth" or "fitHeight" to make FSI Viewer zoom a page to fit the entire width or height of a page. If you do not specify this parameter FSI Viewer displays the entire image.

PreloadForward	
Description	Number of pages to pre-cache
Syntax	Number
Default	4
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Number of subsequent pages to load in advance.

PreloadBackward	
Description	Number of pages to pre-cache
Syntax	Number
Default	2
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Number of preceding pages to load in advance.

PageCacheLimit	
Description	Number of pages to hold in memory
Syntax	Number
Default	8
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.3

Defines the number of pages to hold in memory. Using larger numbers will make FSI Pages consume more memory (RAM) and cached pages will be read from the browser's file cache less frequently.
Use "false" to hold all pages in memory once loaded. This is the default behaviour of previous versions.

PageCacheRemoveTime	
Description	Minimum time each page is being held in memory
Syntax	Number (seconds)
Default	3
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.3

Defines the minimum time each page is being held in memory after it has been viewed the last time. Set this parameter to "0" to remove pages exceeding the "PageCacheLimit" immediately.

PageOverlays	
Description	Defining overlay(s) for all or a specific page
Syntax	XML
Default	—
Context	Pages plug-in node
Version	4.0.0 or above

Define a global page overlay for all pages and/or for individual pages.
A page overlay is a movie clip that loads on top of a page. To specify a default overlay for all pages except for those, you specified an individual overlay for, please add a default="true" attribute to the <overlay> node.
Please see chapter "Page Overlays" for details.

Defining a global page overlay

```
<Plugins>
...
       <Plugin src="pages" >
              <PageOverlays>
              ...
                     <Overlay default="true" file="ov_photo.swf" />
                     <Overlay page="5"  file="myOverlay1.swf" posX="6" posY="12"/>
                     <Overlay page="21" file="myOverlay2.swf" posX="0" posY="98"/>
              ...
              </PageOverlays>
       </Plugin>
...
</Plugins>
```

The example defines the default overlay "ov_photo-swf" and defines individual overlays for pages 5 and 21.

PageOverlayShowZoomed	
Description	Show overlays when zooming pages
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or above

Defines whether to show or hide page overlays when zooming into a page.

MovePages	
Description	Move pages when zooming
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Move the pages when zooming to maximize the space available for FSI Viewer. If set to "false" images will be zoomed in-place.

XMLBase	
Description	Path to XML page data
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"

If you specify this parameter, FSI Pages will try to load XML data for each page from the location specified.
The file path of the XML files must match XMLBase/[image filename].xml

Print	
Description	Enable printing of pages
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.0 or higher

Enable or disable the user to print the content of the current page. Please refer to the chapter "*Printing and Saving Pages*" for details.

PrintResolution	
Description	Max. size of the image to print
Syntax	Number (pixels)
Default	2000
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

Defines the maximum size of the image in pixels to download for printing. Please note that the actual size depends on the size of the image and (with eRez server) on the maximum size defined in the real time template used with FSI Viewer (see parameter *PrintTemplate*).

PrintEffects	
Description	Image effects for the image to print
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

Defines image modification parameters to apply to images used for printing. The value of this parameter depends on the imaging server being used. With FSI Server you can sharpen the image and define the JPEG compression level, e.g. "`effects=sharpen(230)&quality=95`".

PrintTemplate	
Description:	Real time template for the image to print (FSI Server and eRez only)
Syntax	String
Default	fsi
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version:	3.6.0 or higher

Defines the FSI Server profile or eRez real time template used to load the print image. By default FSI Pages uses the default real time template "fsi".

Save	
Description	Enable / disable saving of pages
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

Enables save functionality. Using the save function you can enable the user to open or download a document related to the current page – e.g. the source PDF document of a catalog. Please refer to the chapter "*Printing and Saving Pages*" for details.

SaveDocumentFile	
Description	URL to a PDF document containing all pages
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

URL to a document containing all pages displayed in an FSI Pages instance. This is the URL the user will be redirected to when choosing the "Entire Document" option in the "Save" dialog. Please refer to the chapter "*Printing and Saving Pages"* for details.

SaveURLPrefix and SaveUrlSuffix	
Description	Prefix or suffix for page download URLs
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

Specifies a prefix and/or suffix for the URL FSI Pages opens when the user chooses the "Left Page" or "Right Page" option in the Save dialog. Please refer to the chapter "*Printing and Saving Pages"* for details.

SaveURLTarget	
Description	Target frame for downloading a file related to a page
Syntax	String
Default	_blank
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

HTML frame to open page related files (SaveURL) in.
Please refer to the chapter "*Printing and Saving Pages*" for details.

ForceSaveURL	
Description	URL to use for downloading a file related to a page
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.6.0 or higher

This parameter overrides the download URL specified in the "file" attribute of the page's XML data. Specifying this parameter forces FSI Pages to use the same download URL for all pages.
Please refer to the chapter "*Printing and Saving Pages*" for details.

SaveAllowSavingImages	
Description	Add image download options to the "Save" dialog
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

Set this parameter to "false" if you want to prevent the user from downloading images of pages. In this case the PDF download options will be displayed only.
Please refer to the chapter "*Printing and Saving Pages*" for details.

SaveResolution	
Description	Max. resolution for downloading images
Syntax	Number (pixel)
Default	2000
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

The maximum resolution for images the user can choose to download in the "Save" dialog. Please note that the actual size of the image might be limited by the resolution of the source image or by an imaging server restriction.
Please refer to the chapter "*Printing and Saving Pages*" for details.

SaveEffects	
Description	Image effects for the image to download
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

Defines image modification parameters to apply to images used for downloading. The value of this parameter depends on the imaging server being used.
With FSI Server you can sharpen the image and define the JPEG compression level, e.g. "`effects=sharpen(230)&quality=95`".

SaveTemplate	
Description	FSI Server profile or eRez template to use when downloading images
Syntax	String
Default	fsi
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

The FSI Server profile or eRez real time template to use when downloading images in the "Save" dialog. Please refer to the chapter "*Printing and Saving Pages"* for details.

SaveOptions	
Description	Enable or disable options in the "Save" dialog
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

A comma separated list of options to be presented to the user in the "Save" dialog. The default is presenting all options which corresponds to the following parameter value:

```
document,chapters,both,left,right
```

Using the parameter value: left,right enables saving the left or right page only. Please note that the options might as well not appear, if there is no data available or specified to download for any of these options. Please refer to the chapter "*Printing and Saving Pages"* for details.

Search	
Description	Enable searching a catalog
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

Enables or disables the search interface of FSI Pages. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchDialogWidth and SearchDialogHeight	
Description	Size of the search dialog
Syntax	Number (pixel)
Default	400 (width) / 300 (height)
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

The size of the search dialog in pixel. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchAutoWildCards	
Description	Add wildcards ("*") to keywords automatically
Syntax	Boolean
Default	true
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

By default FSI Pages adds wildcards ("*") to the keywords the user entered into the search input of the search dialog. Set this parameter to "false" if you don't want FSI Pages to add wildcards automatically. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchAutoExactMatch	
Description	Add quotes (") to keywords automatically
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or higher

Setting this parameter to "true" makes FSI Pages add quotes to search requests so that only exact matches will be returned. Setting this parameter implicitly disables the "SearchAutoWildCards" parameter.

SearchParameters	
Description	Adds parameters (modifiers) to the search query
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.1.5 or higher

This parameter provides the possibility to add parameters to the search query within FSI Pages. The parameters need to be provided in the regular HTTP request format: `param1=value1&param2=value2[…]`.

SearchCustomURL	
Description	URL to request the search result from
Syntax	URL
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

You can assign a custom URL to direct search requests to using this parameter. For example you might want to use a database and server side script to return pages related to the search request. This is the default way if you are not using imaging server. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchCustomValue	
Description	A custom value to pass with the search request
Syntax	String
Default	—
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

A custom string to pass to the server along with the search request. This option is useful when using a server other than FSI Server or eRez to return search results. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchQueryTemplate	
Description	Template for the search value
Syntax	String
Default	See description below
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.5.0 or higher

Specifies the template to create the search query value.

When using FSI Server the default templates is:
iptc.caption:`%%searchtext%%%20iptc.fsi_search_data:%%searchtext%%`

This limits the search result to matches in the IPTC Caption and the special IPTC field "fsi_search_data". These are the places FSI Pages Converter adds the search keywords to. In order to search all text meta data fields available you can specify the following template instead: `%%searchtext%%`
Please refer to the FSI Server documentation for additional search options.

When using eRez or another server (parameter "SearchCustomURL"), the default SearchQueryTemplate is:
`%%searchtext%%` (the text the user entered)

Please refer to the chapter "Searching in FSI Pages" for further details.

SearchUseMethodGet	
Description	Use HTTP GET method to request search results
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	4.0.0 or higher

By default FSI Pages requests search results using the HTTP POST method.
Using this parameter you can force FSI Pages to use the HTTP GET method when requesting search results. Please refer to the chapter "*Searching in FSI Pages*" for details.

SearchSortResults	
Description	Sort search results
Syntax	Boolean
Default	True
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	5.0.4

If set to true, the search results are ordered by page number.
If set to false, the search results are listed in the order the imaging server sends them to FSI Pages.

CropLeft, CropTop, CropRight, CropBottom	
Description	Crop page images
Syntax	Number in pixel or percent
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specify the amount in pixel (e.g. "100") or in percent (e.g. "10%") to crop from the specified side of the page image. This parameter is especially useful to remove crop marks from catalogs.
Instead of using the crop amounts "CropLeft" and "CropRight" you might want to use "CropInner" and "CropOuter" for alternating crop margins for even/uneven pages.

CropInner, CropOuter	
Description	Crop page images
Syntax	Number in pixel or percent
Default	0
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Specify the amount in pixel (e.g. "100") or in percent (e.g. "10%") to crop from the inner or outer side of the page image.
These parameters override the "CropLeft" and "CropRight" parameters. "CropInner" is the right side for even pages and the left side for uneven pages. "CropOuter" is the left side for even pages and the right side for uneven pages.

Events	
Description	Enable plug-in events
Syntax	Boolean
Default	false
Context	Pages plug-in node or HTTP query prefixed with "pages_"

Using this parameter you can enable or disable plug-in event notifications e.g. for the JS Bridge plug-in. Please see "Event Notifications and Actions" for details.

FlipCornerSize	
Description	Size of interactive page corners
Syntax	Number
Default	10%
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Specifies the size of the interactive page corners you can use to flip pages. You can either specify the size in pixels or in percent of the page width.
Setting this parameter to "0" disables interactive page flipping by dragging the page corners.

FlipEdgeSize	
Description	Size of interactive page edges
Syntax	Number
Default	5%
Context	Pages plug-in node or HTTP query prefixed with "pages_"
Version	3.5.5 or higher

Specifies the size of the interactive outer page edges you can use to flip pages. You can either specify the size in pixels or in percent of the page width.
Setting this parameter to "0" disables interactive page flipping by dragging the page edges.

NeptuneLabs FSI Viewer

FSI Showcase Add-on

FSI Showcase Add-on integrates a resizable scroll pane to FSI Viewer displaying thumbnails of multiple images. Clicking a thumbnail displays the image in FSI Viewer.
This way a single viewer instance can be used to display hundreds of images presenting thumbnails and optional texts for each image.

Using FSI Showcase

The integration of FSI Showcase is very similar to an ordinary FSI plug-in. To enable the showcase you have to add a plug-in node to your (default) XML configuration file:

Integrating FSI Showcase

```
<Plugins>
...
       <Plugin src="showcase" basecolor="58A8FF"/>
...
</Plugins>
```

Defining Image Lists

The images to be displayed in FSI Showcase have to be defined in an <images> node added to your XML configuration file. The format and options of the collections is the same as the collections for FSI Pages Add-on.

Defining FSI Showcase image nodes

```
<Images>
...
       <Image label="My First Image" file="image1.fsi" />
       <Image label="My Second Image" file="image2.fsi" />
...
</Images>
```

There are three different ways of defining image lists:

1. Creating image nodes referencing external XML configuration files

External XML configuration files

```
<Images>
...
	<Image label="My First Image" file="image1.fsi" />
	<Image label="My Second Image" file="image2.fsi" />
...
</Images>
```

2. Creating image nodes containing complete image configurations

Complete XML configurations

```
<Images>
...
	<Image label="My First Image">
		<FPX>
			<SRC value="image1.fpx" />
		</FPX>
	</Image>

<Image label="My Second Image">
	<FPX>
		<SRC value="image2.fpx" />
		<Width value="8096" />
		<Height value="12300" />
	</FPX>
	<Options>
		<NoNav value="true" />
	</Options>
</Image>
...
</Images>
```

3. Retrieving automatically generated image lists from a server

Complete XML configurations

```
<Plugins>
...
	<Plugin src="showcase" server="..." dir="..." />
...
</Plugins>
```

Method 1 and Method 2 can be freely combined while Method 3 retrieves a complete image list ignoring previously defined <images> nodes.
Please note that the format of image lists is the same as for FSI Pages. The chapter on FSI Pages contains more detailed information on creating and retrieving *image lists*.

Image Order
The order of the images within the thumbnail bar corresponds to the order of the <image> declarations within the <images> section.

Graphical Bookmarks and Product Tours
The Showcase does not re-load images if the image source did not change (e.g. "FPXSrc" and "Effects"). You can therefore use the Showcase to create graphical bookmarks or product tours by specifying the same image source ("FPXSrc") for multiple images and defining different "InitialView" parameters. The Showcase parameter "UseInitialView" should be set to "true" (default) in this case.

FSI Parameters and FSI Showcase
All FSI parameters and FSI Plug-in parameters can be used with FSI Showcase Add-on. The only exception is the plug-in integration which will be described later on. The Showcase Add-on follows the hierarchy displayed below to build the configuration for an image:

Hierarchy of parameter definitions

1. Default Configuration File
2. Showcase Configuration File
3. Image Configuration File
4. Image Node
5. HTTP Query

This way you can define global parameters for all images by defining the parameters in
1 (default configuration)
2 (showcase configuration) or
5 by HTTP query.

Parameters specific to certain images have to be defined in the external image configuration file (3) or in the <image> node of the image list (4).

Due to the hierarchy displayed above parameters declared by query override all other parameter definitions. This can be especially useful if you want to override parameters defined in external configuration files, for example overriding the "MenuAlign" parameter.

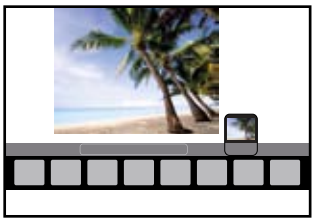
FSI Showcase Layout

The appearance of the Showcase Add-on can be customized by a large number of configuration parameters. The parameters will be described in detail in the sections "Basic Parameters" and "Advanced Parameters".
This section describes general layout considerations and measures.

Parameter Layout:

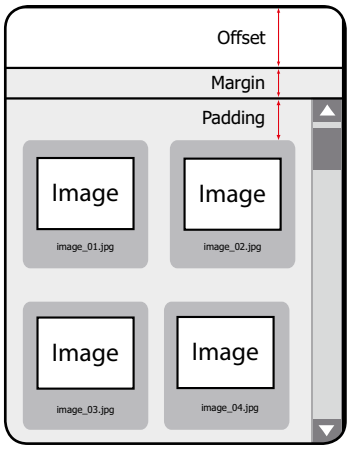
Outside	Inside	Floating
		
Shrinks the size of the viewer by the initial size of the thumbnail bar	*Shrinks the size of the image (view port) by the initial size of the thumbnail bar*	*Floats above the viewer without modifying the viewer or image size*

Parameter Align:

Right	Left	Top	Bottom
			

The figures to the left show the alignment with "Layout" set to "Outside" and "MenuAlign" set to "BL" (Bottom Left). You can achieve various layouts by modifying the 2 Showcase parameters ("Layout" and "Align") and the FSI Viewer parameter "MenuAlign".

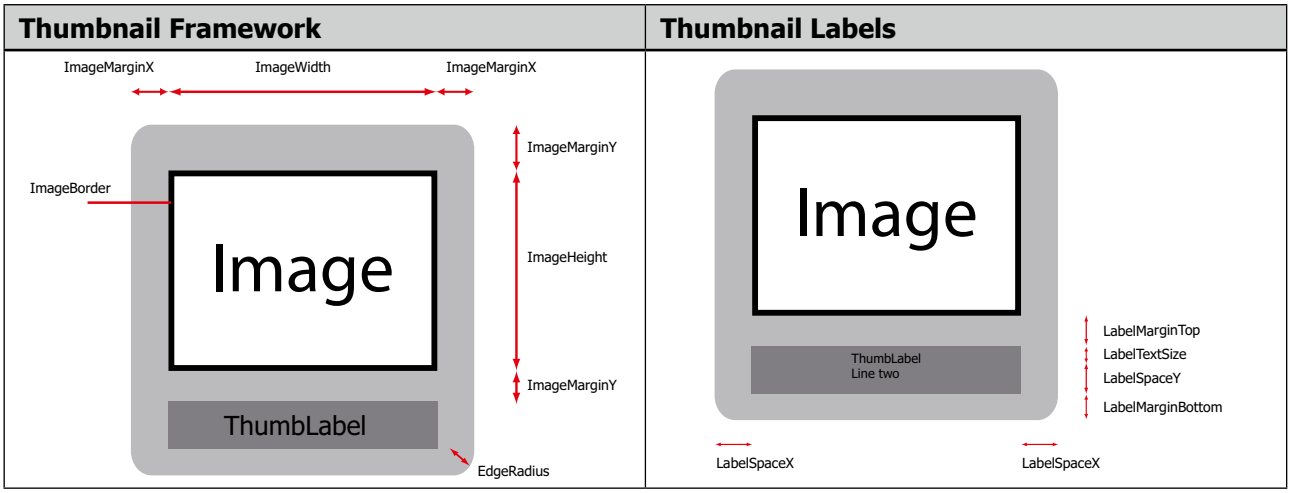
In addition to the layout parameters explained above you can define Offsets, Margins and Paddings for each side of the Showcase. If you use the "Inside" or "Floating" layout you will most likely have to define an offset to avoid parts of the Showcase being hidden below the menu bar.
The actual values depend on the FSI Skin being used. The figure to the right illustrates the parameters "OffsetTop", "MarginTop" and "PaddingTop".
The values for the right, left and bottom side can be set accordingly.



- Use Offset to offset the entire thumbnail bar
- Use Margin to offset the scroll bar and thumbnails
- Use Padding to offset the thumbnails

Using Offset values you can define small thumbnail bars that do not occupy the entire width or height of the viewer.

The chapter "Advanced Parameters" provides a description of parameters which can be used to resize and/or colorize all elements of the thumbnail framework.
If you just want to change the general color scheme you can use the parameter "BaseColor" instead. You can rescale the thumbnail using the parameter "BaseScale".



Using FSI Showcase and FSI plug-ins

You can use additional plug-ins just like with FSI Viewer. There are some rules you should follow when working with additional plug-ins:

- You can not add plug-ins dynamically after FSI Showcase loaded.
  You can on the other hand disable plug-ins for individual images adding an <options> parameter with the name of the plug-in to the corresponding <image> tags of the image list:

Example – Disabling a plug-in inside an <image> definition

Defining FSI Showcase image nodes

```
<Images>
...
       <Image label="No Magnifier" >
              <FPX>
                     <SRC value="image2.fpx" />
                     <Width value="8096" />
                     <Height value="12300" />
              </FPX>
              <Options>
                     <Magnifier value="false" />
              </Options>
       </Image>
...
</Images>
```

- Plug-in parameters defining the initial state of a plug-in do only apply to the first image. The initial state does not get restored when selecting an image in the thumbnail bar. For example if you set the "visible" parameter of the "Magnifier" plug-in to "true" it will be initially visible, but it will not be displayed each time the user selects an image if the user disabled the magnifier glass by clicking the corresponding button.

Showcase Parameters

The Showcase parameters described in the following can be provided in 3 different ways:

a) Attributes of the plug-in tag

Example – Parameter provided as attribute

```
<Plugins>
...
       <Plugin src="showcase" Layout="Outside" />
...
</Plugins>
```

b) Child nodes of the plug-in node

Example – Parameter provided as child nodes

```
<Plugins>
...
       <Plugin src="showcase" >
              <Layout value="Outside" />
              <Align value="Left" />
       </Plugin>
...
</Plugins>
```

c) By HTTP query value prefixed with "showcase_"

Example – Parameter provided by query

```
<Param name="movie" value="fsi.swf?showcase_layout=Outside">
```

The following value types can be used:

Type	Example
Number	"90"
Pixel	"90"
String	"ZoomIn"
URL	"http://www.neptunelabs.com/"
Bool	either "0" / "1" or "true" / "false"
HexColor	"FF00FF"
Color Transition	"FF00FF, FFFFFF"

Alphabetical Index of FSI Showcase Parameters

Basic Parameters

BaseColor	
Description	Base color of the thumbnail bar
Syntax	HexColor
Default	"DDDDDD" (light gray)
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Base color for the thumbnail frame work, the scroll bar and the background of the thumbnail bar. All color values will automatically be derived from this color if not defined separately (see "Advanced Parameters").

BaseScale	
Description	Scale of thumbnail images
Syntax	Number
Default	100
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Scale of thumbnails in percent, e.g. "150" for 150% of the original size.
The user can additionally scale the thumbnails by clicking the corresponding buttons in the drag bar menu. Please note that this value does not change the dimension of thumbnails being retrieved from the server (see ImageWidth and ImageHeight).

Layout	
Description	Defines the way FSI Showcase integrates into the viewer
Syntax	"outside", "inside" or "floating"
Default	"outside"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

- Outside (default)
  The showcase occupies the entire height or width of the viewer instance shrinking the viewer size by the initial size of the thumbnail bar. The thumbnail bar will be placed above all viewer elements when enlarging the thumb bar by default.
- Inside
  The showcase reduces the view port of the viewer by the initial size of the thumbnail bar, leaving the menu elements unchanged. The thumbnail bar will be placed below all viewer elements when enlarging the thumb bar by default.
- Floating
  The showcase floats above the view port without modifying the viewer or the view port size. The thumbnail bar will be placed below all viewer elements when enlarging the thumb bar by default.

Align	
Description	Position of the thumbnail bar
Syntax	"left", "top", "right", "bottom"
Default	"right"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the position of the thumbnail bar. Please note that the orientation of a possible scrollbar can be defined by the parameter "HScroll".

InitialSize	
Description	Initial size of the thumbnail bar
Syntax	Size in pixel or percent or in number of items
Default	"1 item"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the initial size of the thumbnail bar.
You can define the size in absolute pixel values (e.g. "120"), in percent of the viewer size ("50%") or by the number of thumbnails being displayed horizontally (Align="left"/"right") or vertically (Align="top"/"bottom").

If you set the "Layout" parameter to "Outside" or "Inside" this value additionally defines the minimum size of the thumbnail bar the user can achieve by dragging the thumbnail bar.

HScroll	
Description	Display a horizontal scroll bar if required
Syntax	Boolean
Default	false
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

- false
  Arrange thumbnails vertically and use a vertical scroll bar if required

- true
  Arrange thumbnails horizontally and use a horizontal scroll bar if required

DragBar	
Description	Enable/Disable dynamic resizing of the thumbnail bar
Syntax	"Resizable", "Fixed" or "Hidden"
Default	"Resizable"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

- Resizable
  The user is able to resize the thumbnail bar dynamically.
  The width of the splitter bar of the showcase is increased.
- Fixed
  The user is not able to resize the thumbnail bar dynamically.
  The width of the splitter bar of the showcase is normal.
- Hidden
  The user is not able to resize the thumbnail bar dynamically and
  the splitter bar of the showcase is invisible.

DragBarWidth	
Description	Size of the drag bar
Syntax	Number
Default	8
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

A floating point value defining the width of the drag bar of the thumbnail bar.

DragMenu	
Description	Show/Hide the buttons in the thumbnail bar
Syntax	Boolean
Default	true
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Show the buttons of the thumbnail bar enabling the user to resize the thumbnails and the bar itself. This parameter is always "false" if the parameter "DragBar" is "hidden".

DragMenuWidth	
Description	Size of the menu of the drag bar
Syntax	Number
Default	9
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

A floating point value defining the size of the menu inside the drag bar.

UseInitialView	
Description	Use InitialView parameters for thumbnail images
Syntax	boolean
Default	true
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Use the image section defined by the parameter "InitialView" to create the thumbnail image if set to "true". If set to "False" the entire image will be used.

InitialImage	
Description	Image selected at startup
Syntax	Number
Default	1
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

The index of the image selected on startup. Enter the index of the image from 1 to *n* images.

ToolTips	
Description	Show tool tips when hovering above a thumbnail
Syntax	Boolean
Default	true
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Show tool tips when the mouse cursor hovers above a thumbnail. The text presented in the tool tip is the text of the thumbnail.
This option can be useful if you choose to hide the text on the thumbnail or if the thumbnail label is too long to be entirely displayed on the thumbnail itself. Multi-line texts are being presented separated by commas.

MinimizeOnSelect	
Description	Minimize the thumbnail bar on select
Syntax	Boolean
Default	true
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Minimize the thumbnail bar to the initial size when the user selects (clicks) a thumbnail.

TextOutsideFrame	
Description	Display text outside thumbnail frame
Syntax	Boolean
Default	false
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Display the thumbnail label inside the thumbnail frame work (false) or outside the frame work (true). The option only applies to thumbnails in normal state, it does not have an impact on active and selected thumbnails.

ThumbEffects	
Description	Image manipulation parameters applied to thumbnail images
Syntax	String
Default	—
Context	<Options>
Version	3.1.1 or higher

Applies image manipulation effects like "sharpen" or "quality" to all thumbnail images. Please refer to the *FSI Viewer* "*effects*" parameter for a detailed description.

ShowControls	
Description	Show or hide image changing buttons
Default	true
Context	<Options>
Version	3.1.1 or higher

If set to false, the buttons to switch to the next or previous iages are hidden.

ImageList Parameters

If you are using FSI Server or eRez imaging server version 3 or above with FSI Showcase you can additionally use automatically generated showcase configuration files for a specific directory on the server or according to a search query. The generated showcase configuration contains all required data listing images in alphabetical order.

Experienced users might additionally edit the server template files used to generate the image list. The corresponding file is located in the
'WEB-INF/templates/list/' directory of your FSI Server:

Filename	Description
image_list.xml	Creates the <Images> section of the Showcase configuration

You can specify custom templates to create the image list using the "ListTemplate" parameter.

The following parameters have to be passed to the showcase in order to use automatic showcase configurations:

Dir	
Description	Image server Base directory
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Specifies the image directory on the imaging Server the <Images> section will be created for, e.g. 'images/paintings/'

Query	
Description	Search query (eRez server only)
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Specifies a search query to be used to generate the showcase, e.g. "foo*".
Can be used alternatively to the "dir" parameter.

Server	
Description	Path to imaging server
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Specifies the URL of the imaging server to generate the configuration. E.g. 'http://mydomain.com/fsi/'

This parameter can be omitted if the Showcase has been installed to the default location on the imaging server.

Please note: Loading configurations across domain boundaries will fail due to a security restriction of the Adobe Flash™ plug-in.

ListTemplate	
Description	Template to retrieve image lists
Syntax	String
Default	"fsi/image_list.xml"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.2.6

Specifies the XML template to be used to retrieve the image list from imaging server. You can use custom list templates to retrieve image configurations files from imaging server, e.g. to including HotSpot definitions contained in IPTC data of the images.

EmptyImages	
Description	Define blank images (place holders)
Syntax	String
Default	---
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.3.0 or higher

Using this parameter you can add place holders to your image collection.
The value of the parameter must contain one or more image indices the place holders shall occupy.
E.g. `<EmptyImages value="2,10" />` adds place holders at position 2 and 10.
Alternatively you can add `<image empty="true" />` nodes to your image collection.

Advanced Parameters

The following parameters can be used to customize the appearance of the thumbnails and the thumbnail bar. Please note that you can easily change the color scheme by defining a BaseColor rather than defining all colors individually. To change the overall scale of thumbnails you can use the basic parameter "BaseScale".

Background	
Description	Color of the background of the thumbnail bar
Syntax	HexColor
Default	Defined by parameter BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Color of the background of the thumbnail bar.

BackgroundAlpha	
Description	Opacity of the background of the thumbnail bar
Syntax	Number
Default	100
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the opacity of the thumbnail bar background from "0" (invisible) to "100" (opaque).

ScrollBaseColor	
Description	Base color of the scroll bar
Syntax	HexColor
Default	Derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines a color for the scroll bar.

ScrollTrack	
Description	Color of the track bar of the scroll bar
Syntax	HexColor
Default	Derived from ScrollBaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines a color for the track area of the scroll bar.

ScrollArrow	
Description	Color of the arrows of the scroll bar
Syntax	HexColor
Default	Derived from ScrollBaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines a color for the arrows on the buttons of the scroll bar. Please note that the same color applies to the labels of the drag bar menu.

DragBarHighlight	
Description	Highlight color of the drag bar
Syntax	HexColor
Default	Derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.5.0 or higher

Defines the highlight color of the drag bar.

DragBarShadow	
Description	Shadow color of the drag bar
Syntax	HexColor
Default	Derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.5.0 or higher

Defines the shadow color of the drag bar.

DragBarRoundBevel	
Description	Use a round or small bevel for the drag bar
Syntax	Boolean
Default	true
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.5.0 or higher

Use a large, round bevel or a small, rectangular bevel when drawing the drag bar.

ImageWidth and ImageHeight	
Description	Dimension of thumbnail images
Syntax	Pixel
Default	45
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the maximum physical dimension of thumbnail images in pixel. The actual size of thumbnail images depends on the aspect ratio of an image.

ThumbspacingX and ThumbspacingY	
Description	Space between thumbnails
Syntax	Pixel
Default	4
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

The spacing between the thumbnails in the thumbnail bar in horizontal and vertical direction.

ImageMarginX and ImageMarginY	
Description	Space around thumbnail image
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Space between image border and thumbnail border left/right of the image (ImageMarginX) and above/below the image (ImageMarginY).
If you choose round edges (parameter "EdgeRadius") this value will automatically be increased if required to ensure that the image does not overlap the thumbnail edge.

ImageBorderWidth	
Description	Thickness of the border around the thumbnail image
Syntax	Pixel
Default	1
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the thickness of the border around the thumbnail image.
Setting this parameter to "0" disables the image border.

ImageBorderHighlight	
Description	Highlight color of the border around the thumbnail image
Syntax	HexColor
Default	Derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the color of the highlighted section of the image border (left and top).

ImageBorderShadow	
Description	Shadow color of the border around the thumbnail image
Syntax	HexColor
Default	Derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the color of the darkened section of the image border (right and bottom).

ThumbFace	
Description	Color of the thumbnail face
Syntax	HexColor or color transition
Default	Color transition derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the color of the thumb face.
This value can be either defined as a solid color (e.g. "00FF00") or as a linear color transition from top/left to bottom/right by entering two color values separated by comma (e.g. "FF0000, 00FF00").

The corresponding parameters for active and selected state are: ThumbActiveFace and ThumbSelectedFace.

ThumbBorderWidth	
Description	Thickness of the thumbnail border
Syntax	Pixel
Default	1
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Thickness of the border around the thumbnail. Enter "0" to disable the border.

The corresponding parameters for active and selected state are: ThumbBorderActiveWidth and ThumbBorderSelectedWidth.

ThumbBorderHighlight	
Description	Highlight color of the thumbnail border
Syntax	HexColor
Default	Color derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the color of the highlighted section of the thumbnail border (left and top).
The corresponding parameters for active and selected state are:
ThumbBorderActiveHighlight and ThumbBorderSelectedHighlight.

ThumbBorderShadow	
Description	Shadow color of the thumbnail border
Syntax	HexColor
Default	Color derived from BaseColor
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the color of the darkened section of the thumbnail border (right and bottom).

The corresponding parameters for active and selected state are:
ThumbBorderActiveShadow and ThumbBorderSelectedShadow.

EdgeRadius	
Description	Radius of rounded thumbnail edges
Syntax	Pixel
Default	6
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines a radius for drawing rounded edges of the thumbnail frame.
Use "0" for straight edges and positive values for rounded edges.

OffsetTop	
Description	Offset from the top of the viewer instance
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the offset from the top of the viewer instance to the top of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

OffsetBottom	
Description	Offset from the bottom of the viewer instance
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the offset from the bottom of the viewer instance to the bottom of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

OffsetLeft	
Description	Offset from the left of the viewer instance
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the offset from the left of the viewer instance to the left of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

OffsetRight	
Description	Offset from the right of the viewer instance
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the offset from the right of the viewer instance to the right of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

MarginTop	
Description	Margin from the top of the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the margin in pixels from the top of the thumbnail bar to the content and scrollbar of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

MarginBottom	
Description	Margin from the bottom of the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the margin in pixels from the bottom of the thumbnail bar to the content and scrollbar of the thumbnail bar of the showcase. Please see the "Showcase measurements" figures for details.

MarginLeft	
Description	Margin from the left of the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the margin in pixels from the left of the thumbnail bar to the content and scrollbar of the thumbnail bar of the showcase. Please see the "Showcase measurements" figures for details.

MarginRight	
Description	Margin from the right of the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the margin in pixels from the right of the thumbnail bar to the content and scrollbar of the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

PaddingTop	
Description	Padding of thumbnails inside the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the top padding of thumbnails inside the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

PaddingBottom	
Description	Padding of thumbnails inside the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the bottom padding of thumbnails inside the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

PaddingLeft	
Description	Padding of thumbnails inside the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the left padding of thumbnails inside the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

PaddingRight	
Description	Padding of thumbnails inside the thumbnail bar
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

This value defines the right padding of thumbnails inside the thumbnail bar of the showcase.
Please see the "Showcase measurements" figures for details.

HoverZoom	
Description	Zoom thumbnails "onMouseOver"
Syntax	Number
Default	100 (disabled)
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Zoom thumbnails when the user points at a thumbnail.
The default value is "100" (no effect).
Possible values are 100% (no effect) to 250%.

HoverRotation	
Description	Rotate thumbnails "onMouseOver"
Syntax	Number
Default	0 (disabled)
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Rotate thumbnails when the user points at a thumbnail.
The default value is "0" (no effect).
Possible values are 0 (degree) to 360 (degree).

The following parameters can be used to customize the text displayed on the thumbnails.

LabelTextSize	
Description	Text size of the thumbnail labels
Syntax	Pixel
Default	0 (no text)
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Text size in pixel for the text presented below the thumbnail image.
Use "0" to disable text and positive values (e.g. "8") to enable texts on the thumbnails.

LabelMode	
Description	Content of the thumbnail labels
Syntax	Number
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Obsolete. Please use LabelContent and LabelTemplates parameters described below instead.

LabelContent	
Description	Content of thumbnail labels
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

The specified string serves as a label for the thumbnail images in the thumbnail bar. The string may contain *basic HTML tags* to format the text. Additionally you can add templates being replaced at runtime. Please see "LabelTemplates" parameter below for details.

LabelTemplates	
Description	Runtime Templates for thumbnail labels
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

FSI Showcase add-on replaces the specified templates in the thumbnail labels (parameter "LabelContent") with the value specified in the image's configuration at runtime. This is: any configuration value (specified in _default.fsi, image.fsi or by HTTP Query) can be integrated into the thumbnail label. To do so, you need to
a) Add a corresponding template to the "LabelContent" parameter value
b) Add the template to the "LabelTemplates" value

Each template needs to be entered as an XML node, while the node name is the name of the full configuration parameter name like you would use the parameter in the HTML query for FSI Viewer, e.g. `<FPXWidth/>` or `<Showcase_Dir/>`.

Thumbnail labels with runtime templates

```
<Plugin src="showcase">
...
<LabelContent><b><ImageIndex/>. <ImageLabel /></b><br/>Size: <FPXWidth/> x <FPXHeight/>
px</LabelContent>
<LabelTemplates value="FPXWidth,FPXHeight" />
...
</Plugin>
```

The resulting label with this configuration looks like this:

1. SomeImageName.tif
Size: 2048 x 1024 px

The two templates <ImageIndex/> and <ImageLabel /> are built-in templates containing the numeric index of an image and the "label" attribute of the <image> node of the image list. eRez users might want to add templates adding IPTC data served by eRez server to the thumbnail labels. Please create your configuration file based on the FSI Showcase preset "fsi/config/showcase_presets/showcase_info.fsi" in this case.

LabelPrefix	
Description	Prefix applied to thumbnail texts
Syntax	String
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

A prefix string applied to all thumbnail labels or "false" for no prefix.

LabelTextWidth and LabelTextHeight	
Description	Horizontal or vertical space for thumbnail labels
Syntax	Pixel
Default	—
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

You can use this value to increase the horizontal or vertical space available for thumbnail labels. In case the thumbnail label text does not fit into the available space, the label will be displayed cropped, but the entire lable is available in the thumbnail‘s tool tip.

LabelTextAlign	
Description	Text alignment for all thumbnail labels
Syntax	left, center or right
Default	left
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Align the thumbnail labels left, centered, or right in the availabel space for label texts. This value has no effect if the contents of a label contains
`<p align="[alignment] "></p>` tags.

LabelTextFont	
Description	Font used for thumbnail label texts
Syntax	String
Default	"_sans"
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Defines the font used to display thumbnail labels. This value has no effect if the contents of a label contains `<font face="[fontname] "></font>` tags. Please note that the font will only be reflected, if the font is available on the user's system. Otherwise a similar font will be displayed.

LabelTextColor	
Description	Text color of thumbnail labels
Syntax	HexColor
Default	000000
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

A 6-digit hexadecimal color value defining the text color of thumbnail labels. This value has no effect if the contents of a label contains `<font color="[colorvalue] "></font>` tags.

LabelMarginTop and LabelMarginBottom	
Description	Margin above/below the thumbnail text
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

You can use this value to add additional space above (LabelMarginTop) or below (LabelMarginBottom) the thumbnail text.

LabelOffsetTop	
Description	Adjust vertical position of the thumbnail text
Syntax	Pixel
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

You can use this value to adjust the vertical position of the text area of the thumbnail. Negative values move the text area up while positive values move the text area down.

LabelOnRight	
Description	Display thumbnail labels right of the image
Syntax	Boolean
Default	false
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	4.1.3

Set this parameter to "true" to display thumbnail labels right of the images instead of below the thumbnail images.

LabelOffsetLeft	
Description	Adjust horizontal position of thumbnail text
Syntax	Pixel
Default	10
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	4.1.3

You can use this value to adjust the horizontal position of the text area of the thumbnail. Negative values move the text area to the left while positive values move the text area to the right.

PreloadThumbs	
Description	Load all thumbnail images on startup
Syntax	Number
Default	0
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"

Setting the value to "0" makes the showcase load thumbnails on-demand.
Setting positive values enables the pre-loader which loads all images on startup The given value specifies the number of images to preload simultaneously.

Please note: Be careful when activating this option especially with automatically generated image lists or large image lists as it might cause higher server loads and increased (and possibly unnecessary) network traffic.

CropThumbnails	
Description	Crop images to match thumbnail dimension
Syntax	Boolean
Default	false
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.5.0 or higher

Automatically crop images to match the dimensions of the thumbnails.

KeepImageSection	
Description	Preserve the view when selecting thumbnails
Syntax	Boolean
Default	false
Context	Showcase plug-in node or HTTP query prefixed with "showcase_"
Version	3.5.0 or higher

Usually FSI Showcase displays the entire image (or the image section specified by the parameter "InitialView") when selecting a thumbnail. Set this parameter to "true" to if you want to preserve the image section you are currently viewing when selecting another image.

Plug-in Reference

This part of the documentation lists all standard FSI Viewer plug-ins. The purpose of FSI Viewer plug-ins is integrating functionality at runtime that might not be required in general. This way users need to download only that part of FSI Viewer code necessary to provide the required functionality. Additionally the plug-in system provides the possibility to add functionality required by an individual customer or by a certain group of customers. Please contact your FSI Viewer dealer in case you are missing any kind of functionality.

Plug-in BackgroundImage

Plug-in Target:
Viewer background

Plug-in Location:
/plugins/backgroundimage.plg

Function:
Adds a background image to FSI Viewer

Syntax:

```
<Plugin src="backgroundimage"
FPXSrc="images/bgimage.tif" />
```

This plug-in adds a background image to FSI Viewer. Usually
background images are used with FSI Pages only, as with FSI Viewer and FSI Showcase most of the background image is covered by the zoomable image.
The plug-in crops the specified image so that the entire area of FSI Viewer instance is filled with the background image.

Plug-in Parameters

FPXSrc	
Description	Relative path to the image on your imaging server
Syntax	String
Default	—
Context	Plug-in tag

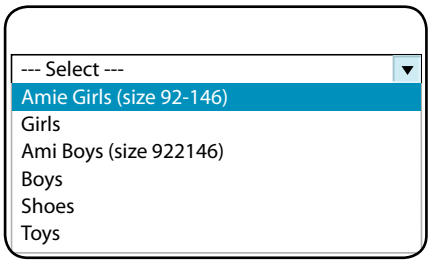
The path to an image on your imaging server to be used as a background.
If you use a relative path (recommended) the FSIBase parameter will be used to get the absolute path to the image.

Effects	
Description	Optional effect parameters to pass to the imaging server
Syntax	String
Default	—
Context	Plug-in tag

Like with the *Effects parameter* for FSI Viewer itself you can optionally pass image effect parameters to your imaging server that affect the appearance of the background image.

Plug-in Chapters

Plug-in Target:
User interface of FSI Pages



Plug-in Location:
/plugins/chapters.plg

Function:
Adds a select box to access parts of a catalog directly.

Syntax:

```
<Plugin src="chapters"/>
```

This plug-in adds a select box to the user interface of FSI Pages. The user can directly access sections of a catalog by selecting the corresponding item in the select box. You can assign colors and different levels to each chapter using simple XML based data describing the document structures.

Example: XML data for the Chapters plug-in

```
<Indexdata>
	<Index label="First Page" page="1" color="FFFF00" />
	<Index label="Chapter 1" page="10" color="00FF00">
		<Index label="Chapter 1.1" page="20" color="00FF00"/>
		<Index label="Chapter 1.2" page="30" color="00FF00"/>
		<Index label="Chapter 1.3" page="40" color="00FF00"/>
	</Index>
	<Index label="Chapter 2" page="50" color="0000FF"/>
	<Index label="Chapter 3" page="60" color="CCCCCC"/>
	<Index label="Chapter 4" page="70" color="FF0000"/>
</Indexdata>
```

Plug-in Parameters

IndexData	
Description	XML document structure data
Syntax	XML Data
Default	---
Context	Plug-in attribute

XML Data describing the document structure. Add the XML data as a child node of the "Chapters" plug-in node. Alternatively you can specify the XML data using an external XML file.

IndexDataFile	
Description	URL to an external XML file describing the document structure
Syntax	String
Default	---
Context	Plug-in attribute

Alternatively to the "IndexData" parameter you can use an external XML file to provide the XML document structure data required for this plug-in. The XML file must be located in the same domain as FSI Viewer to avoid cross-domain security restrictions. If you enter a relative path, FSI Viewer loads the file from the subdirectoy "pages_url" of your FSI Viewer configuration directory ("FSIBase")

If you store XML files in the directory "/[fsi]/config/pages_xml/" you therefore only need to specify the file name of the XML file to load.
Please note: You need to use an external XML file in UTF-8 format when using special language specific characters (characters >= 0x80).

PageNumbers	
Description	Display a starting and ending pages of chapters
Syntax	Boolean
Default	true
Context	Plug-in attribute

Add the starting and ending page of a chapter to the chapter label.

ComboLabelBackgroundAlpha	
Description	Background opacity of the select box background
Syntax	Number
Default	10
Context	Plug-in attribute

Opacity of the select box background from 0 (transparent) to 100 (opaque).

ColoredIndex	
Description	Display colored index markers in the select box
Syntax	Boolean
Default	true
Context	Plug-in attribute

Display the color specified in the document structure XML data in front of the chapter labels.

IndexMarginWidth	
Description	Width of the colored page index
Syntax	Number
Default	0
Context	Plug-in attribute

Floating point value specifying the width of the colored index located on the outer page margin in percent of the page width. Enter 0 to disable the colored page margin.
Please note: This value requires setting "ColoredIndex" to true as well.

Width	
Description	Width of the select box
Syntax	Pixels
Default	200 (depending on skin)
Context	Plug-in attribute

Usually the FSI Pages skin defines the width and position of the select box. Alternatively you can specify a fixed with in pixels using this parameter.

Sort	
Description	Sort index items by page number
Syntax	Boolean
Default	true
Context	Plug-in attribute
Version	4.1.5 or above

By default the plug-in sorts the index items ascending by page number.
Set this parameter to "false" to use the order in which the index items are listed in the XML file.

Update	
Description	Update the selected index on page flip
Syntax	Boolean
Default	true
Context	Plug-in attribute
Version	4.1.5 or above

By default the plug-in automatically selects the matching index item after a page flip. Set this parameter to "false" to display the first index item always, e.g. to make the plug-in display "Go to section …" always.

ValidateData	
Description	Use valid index items only
Syntax	Boolean
Default	true
Context	Plug-in attribute
Version	4.1.5 or above

By default the plug-in does not display items that refer to page numbers higher than the last page or lower than first page. Set this parameter to display all index items, even if refer to non-existing pages.

Index Item Attributes
Each <index> node represents an item in the chapters drop down referring the user to a page in your document. The following attributes need to or can be added to each <index> item:

Attribute	Description
page	Index of the page to refer to (**required**)
label	Title for a chapter (**required**)
file	File to download for a chapter (from version 4.0.0) See chapter Saving Pages for details.
color	6-digit hexcolor for use with "ColoredIndex" parameter
musicSong	An absolute or relative URL to an MP3 file to play for this chapter (from version 4.0.7). **Please note** that you need to add the plug-in "music" as well in this case.

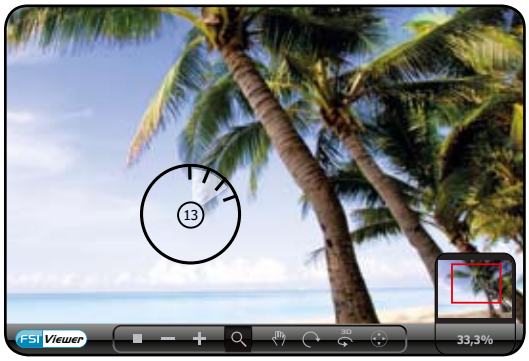
Plug-in ClockProgress

Plug-in Target:
User interface

Plug-in Location:
/plugins/clockprogress.plg

Function:
Replacement for default progress bar

Syntax:
`<Plugin src="clockprogress" />`



The "ClockProgress" plug-in replaces the default progress bar with a round progress indicator.

Plug-in Parameters

The clockprogress plug-in can be customized by adding attributes to the <plugin> tag.
E.g. `<Plugin src="clockprogress" alpha="20"/>`

The following attributes can be used to customize the clockprogress plug-in:

Alpha	
Description	Opacity of progress indicator
Syntax	Number
Default	50
Context	Plug-in attributes

Defines the opacity of the progress indicator. Possible values range from 0 to 100.

0 transparent
100 opaque

Size	
Description	Size of progress indicator
Syntax	Number
Default	60
Context	Plug-in Tag

Defines the overall size of the progress indicator in pixel.

Color1	
Description	Color of the area representing the amount of pending data
Syntax	HexColor
Default	000000
Context	Plug-in attributes

A 6-digit hexadecimal number defining the color of the area representing the amount of pending data.

Color2	
Description	Color of the area representing the amount of data already loaded
Syntax	HexColor
Default	FFFFFF
Context	Plug-in attributes

A 6-digit hexadecimal number defining the color of the area representing the amount of data already loaded.

LineColor	
Description	Line color
Syntax	HexColor
Default	000000
Context	Plug-in attribute

A 6-digit hexadecimal number defining the color of the lines.

TextColor	
Description	Text color
Syntax	HexColor
Default	000000
Context	Plug-in attribute

A 6-digit hexadecimal number defining the color of the text.

posX	
Description	Horizontal Position
Syntax	String / Number
Default	Depends on MenuAlign parameter
Context	Plug-in attribute

Defines the horizontal position of the progress indicator.

Possible values:

C	Centered horizontally
Number > 0	Offset from left
Number < 0	Offset from right

posY	
Description	Vertical position
Syntax	String / Number
Default	Depends on MenuAlign parameter
Context	Plug-in attribute

Defines the vertical position of the progress indicator.

Possible values:

C	Centered vertically
Number > 0	Offset from top
Number < 0	Offset from bottom

Plug-in ColorAdjust

Plug-in Target:
User interface

Plug-in Location:
/plugins/coloradjust.plg

Function:
Adjusts the color level of images

Syntax:
```
<Plugin src="coloradjust" />
```

Plug-in Parameters

AutoHideDialog	
Description	Auto Hide Dialog
Syntax	Boolean
Default	true
Context	ColorAdjust
Version	5.0.0

Hides the interface dialog when changing values.

Blue	
Description	Blue value
Syntax	Float
Default	100
Context	ColorAdjust
Version	5.0.0

Amount of blue.

Brightness	
Description	Brightness of image
Syntax	Float
Default	100
Context	ColorAdjust
Version	5.0.0

The image brightness.

Green	
Description	Green value
Syntax	Float
Default	100
Context	ColorAdjust
Version	5.0.0

Amount of green.

MenuOffset	
Description	MenuOffset
Syntax	Number (Pixels)
Default	—
Context	ColorAdjust
Version	5.0.0

Defines the menu offset.

Red	
Description	Red value
Syntax	Float
Default	100
Context	ColorAdjust
Version	5.0.0

Amount of red.

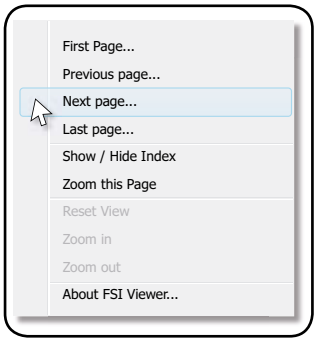
SetValuesForEachImage	
Description	Save Values for each image seperately
Syntax	Boolean
Default	false
Context	ColorAdjust
Version	5.0.0

If enabled, the color values can be set for each image. If disabled (default) the values are used for all images.

ShowUI	
Description	Show Interface
Syntax	Boolean
Default	true
Context	ColorAdjust
Version	5.0.0

If disabled, the user interface will be hidden, enabling the effects.

Plug-in ContextMenu



Plug-in Target:
User interface

Plug-in Location:
/plugins/contextmenu.plg

Function:
Adds FSI Viewer related items to the Flash player's context menu

Requirements:
Flash player 7 or above

Syntax:

```
<Plugin src="contextmenu" />
```

This plug-in does not provide any parameters. You can on the other hand translate the menu items or additional items. The example below shows the built-in configuration of this plug-in.

Remove Drag-Button

```
<Plugin src="contextmenu" >
            <MenuItems>
                  <!-- FSI PAGES ONLY -->
                  <Item label="First Page..." action="FirstPage" />
                  <Item label="Previous page..." action="PreviousPage" />
                  <Item label="Next page..." action="NextPage" />
                  <Item label="Last Page..." action="LastPage" />
                  <Separator />
                  <Item label="Show / Hide Index" action="ShowHidePageIndex"/>
                  <Item label="Zoom this Page" action="ZoomPage"/>
                  <Separator />
                  <!-- FSI SHOWCASE ONLY -->
                  <Item label="Previous Image" action="PreviousImage"/>
                  <Item label="Next Image" action="NextImage"/>
                  <Separator />
                  <!-- FSI VIEWER -->
                  <Item label="Reset View" action="Reset"/>
                  <Item label="Zoom  in" action="ZoomIn"/>
                  <Item label="Zoom  out" action="ZoomOut"/>
            </MenuItems>
</Plugin>
```

Plug-in Parameters

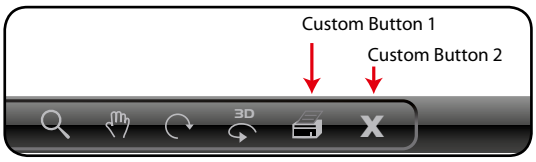
MenuItems	
Description	Refers to the XML describing the context menu entries
Syntax	String
Default	—
Context	Plug-in tag
Version	5.0.0

Defines the XML snippet describing the context menu. Please refer to the *FSI Contrib package* if you want to modify or translate the context menu.

Plug-in CustomButton

Plug-in Target:
User interface

Plug-in Location:
/plugins/custombutton.plg



Function:
Adds custom buttons to the menu bar

Syntax:

```
<Plugin src="CustomButton" buttons="btn1,btn2" />
```

The "CustomButton" plug-in can be used to integrate one or more custom buttons into the menu bar of FSI viewer. Clicking a button invokes the URL specified in the plug-in parameter. This way you can integrate buttons performing a JavaScript call or opening up a web page.

Plug-in Parameters

MenuOffset	
Description	Offset in pixel to previous menu elements
Syntax	Number in pixel
Default	0
Context	Plug-in tag

Defines an offset of all custom buttons in pixels to elements of the menu bar.

Buttons	
Description	Comma separated list of unique identifiers
Syntax	String
Default	—
Context	Plug-in tag

Enter a list of unique identifiers separated by commas. Each identifier represents a new button that can be configured using the parameters below.
E.g. "btn1,btn2" adds two custom buttons. To specify parameters for these buttons you have to prefix the parameters by the unique identifier plus, e.g. "btn1.Url" defines the URL to use for the button named "btn1".

[ID].Offset	
Description	Offset of the button in pixel to previous (custom) buttons
Syntax	Number in pixels
Default	0
Context	Plug-in tag

Specifies an offset for the button named "[ID]" in pixel to previous (custom) buttons.

[ID].URL	
Description	Url to open on click
Syntax	String
Default	—
Context	Plug-in tag

Specifies an URL to open when the user clicks the button named "[ID]".
You can specify a URL to a web page or alternatively a script call,
e.g. "javascript: void alert("foo");"
Please note: You can use templates to be replaced by FSI Viewer at runtime. The button appears disabled if no URL is specified.

[ID].Frame	
Description	HTML frame to open the specified URL in
Syntax	String
Default	_self
Context	Plug-in tag

Specifies the HTML frame in which to open the specified URL for the button named "[ID]". Besides using an identifier of a frame inside your HTML frame set you can as well use one of the predefined values "_self", "_top", "_blank" or "_parent".

[ID].ToolTip	
Description	Tool tip text
Syntax	String
Default	—
Context	Plug-in tag

Specifies the tool tip to be displayed when the user places the cursor above the custom button named "[ID]".

[ID].LabelFrame	
Description	Symbol displayed on the button
Syntax	Number
Default	1
Context	Plug-in Tag

Specifies the symbol to be displayed on top of the custom button named "[ID]".
The predefined symbols are:

GFX	ID	Example
	1	Information
	2	Close
	3	Resize or Restore
	4	Open window
	5	Print

You can add your own graphics to be used as a label for custom buttons. Please refer to the *FSI Contrib package* for a sample and instructions how to create your own custom button icons. You will need Adobe Flash™ editor to add custom icons.

Using Templates in URL parameters

Starting from FSI version 3.2.3 you can use templates when defining URLs for custom buttons. These templates will be replaced by FSI Viewer when the user clicks a custom button. This way you can retrieve parameters from FSI Viewer dynamically and pass the values to a JavaScript function or a server side script.

Templates in CustomButton URLs

```
<Plugin src="custombutton" buttons="btn1" btn1.URL="javascript:alert('$$cfg.FPXWidth$$
x $$cfg.FPXHeight$$')" />
```

The example above outputs the size of the source image via JavaScript.

The list below shows available templates for custom button URLs:

Template in URL	Replaced by...
`$$cfg.[Name]$$`	The value of the FSI Viewer parameter [*name*]. Be sure to use the long parameter names only (e.g. "**`FPXWidth`**" instead of "**`Width`**"). You can retrieve parameter values retrieved from the imaging server using this template as well. E.g. "**`$$cfg.FPXWidth$$`**" returns the size of the source image – whether specified manually or retrieved from the imaging server.
`$$View$$`	A string representing the current image section in FSI Viewer (see Parameter "*InitialView*" for details)
`$$URLView$$`	The complete URL to the image currently displayed in FSI Viewer. The resulting string contains the templates "**`$$width$$`**", "**`$$height$$`**" and "**`$$template$$`**" (FSI Server and eRez only) which need to be replaced in JavaScript or server side script.
`$$URLScene$$`	Same as above, but representing the entire scene image.
`$DownloadRatio$$`	Aspect ratio representing the factor "width/heigth". You can use this value to calculate the width of the image to download.
`$$Zoom$$`	Current magnification relative to the initial size of the image.
`$$Selection$$`	A string representing the current selection (**requires plug-in SelectFrame**)
`$$Measure$$`	A string representing the current measurement (**requires plug-in Measure**)

Plug-in Fullscreen

Plug-in Target:
User interface

Plug-in Location:
/plugins/fullscreen.plg

Function:
switch to fullscreen mode

Syntax:
```
<Plugin src="fullscreen"/>
```

If using the object tag when embedding FSI Viewer, please note the parameter allowfullscreen has to be set as shown

HTML integration without considering old browsers

```
<Object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.
cab#version=6,0,65,0" width="[Width]" height="[Height]">
      <PARAM name="movie" VALUE="[URL and Parameter]">
      <PARAM name="bgcolor" VALUE="[Background color]">
      <PARAM name="menu" VALUE="false">
      <PARAM name="allowfullsreen" VALUE="true" />"
</Object>
```

Please note, that when using fullscreen the plugin, the Adobe Flash Player prohibits keyboard entries.

Plug-in History

Plug-in Target:
User interface

Plug-in Location:
/plugins/history.plg

Function:
History functionality to jump to previously viewed image sections

Syntax:

```
<Plugin src="history" length="25" />
```

The "History" plug-in extends the user interface by 2 buttons, "History Back" and "History Forward".
Using these two buttons the user is able to navigate through image sections previously viewed.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu buttons
Syntax	Number
Default	0
Context	Plug-in Tag

Specifies the space in pixels left of the menu button(s) of the plug-in.

Length	
Description	Length of the history list
Syntax	Number
Default	10
Context	Plug-in tag

The number of image sections that will be stored in the view history from 1 to 99.
If the list exceeds the amount of entries defined by the "length" parameter, the first entry of the list will be replaced.

Loop	
Description	Loop the history list
Syntax	Boolean
Default	false
Context	Plug-in tag

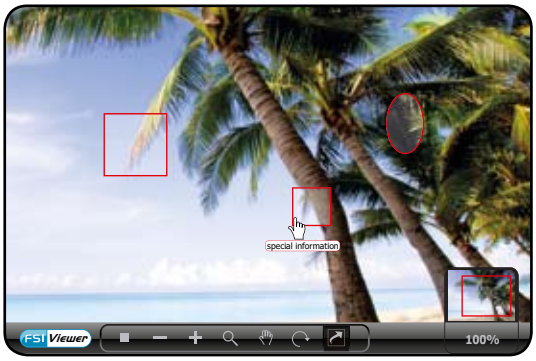
If this value is set to "true" the viewer will display the first viewed section when exceeding the last entry of the history and the last entry when stepping before the first entry of the history list.

Providing the default value "false", the history will not be looped, this is the history list stops at the first and the last position.

SmallButtons	
Description	Size of menu buttons
Syntax	Boolean
Default	true
Context	Plug-in tag

If this value is set to "false" the buttons of this plug-in will be displayed in the default button size. Otherwise the buttons appear slightly smaller than other buttons.

Plug-in HotSpots



Plug-in Target:
User Interface and Zoom Area

Plug-in Location:
/plugins/hotspots.plg

Function:
By defining clickable areas (hotspots) the user is able to interact with the image by clicking on areas inside the zoomable image.

Syntax:
```
<Plugin src="hotspots" />
```

Enterprise editions of FSI Viewer contain this plug-in by default. For all other editions this plug-in can be optionally obtained.

The "HotSpots" plug-in adds an image map like functionality to the FSI Viewer.
The plug-in adds a button to the menu bar which allows the user to show or hide the HotSpots.
The clickable areas are defined via XML inside the image specific *.fsi file and enable an increased interaction with the Viewer.

There are 4 different combinable actions that can be assigned to each hotspot:

Display a tool tip
Zoom to the area defined by the hotspot
Open a HTML page
Restart the Viewer with a different image / configuration

Adding Hotspot Functionality to your Images

Integrating the Plug-in

To integrate the HotSpot plug-in into the viewer you have to add the following line to the <Plugins> section of either the _default.fsi file or to an image specific .fsi configuration file:

```
<Plugin src="hotspots" />
```

Defining the HotSpots Section

To define HotSpots for an image you have to add an XML-section to your image specific .fsi file. The definition syntax is similar to image maps for static images inside a HTML page.
Each HotSpot is being defined by a single XML node inside the <HotSpots> group.

The basic structure of a HotSpot definition is the following:

HotSpot definition

```
<Plugin src="hotspots" visible="true">

      <HotSpots>
            <circle spot="" tip="" url="" />
            <rect   spot="" tip="" url="" />
      </HotSpots>

</Plugin>
```

Alternatively to adding the hotspot XML section as a child node of the HotSpot <plugin> node, you can add the XML section as a root child in the same level as <FPX> or <Options> in a configuration file.

Plug-in Parameters

Visible	
Description	Initial visibility of the HotSpots
Syntax	Bool
Default	false
Context	Plug-in attribute or <HotSpots> group tag

Defines the initial visibility of HotSpots.
The HotSpot button state will be set accordingly.

Alpha	
Description	Defines the opacity for all HotSpots
Syntax	Number between 0-100
Default	100
Context	Plug-in attribute or <HotSpots> tag

Defines the opacity for all HotSpots from 0 (invisible) to 100 (opaque).

BaseURL	
Description	Prefix for relative HotSpot URLs
Syntax	String
Default	—
Context	Plug-in attribute or <HotSpots> group tag

Defines a prefix that will be applied to all HotSpots containing relative URLs.
This way you can omit e.g. the domain in URL-attributes of subsequent HotSpots.

Events	
Description	Enable JSBridge events for HotSpots
Syntax	Boolean
Default	false
Context	HotSpots
Version	5.0.4

Enable to fire event notifications for HotSpots via JSBridge.
The following events will be fired:

1) hotspots_MouseOver
2) hotspots_MouseOut
3) hotspots_MousePress
4) hotspots_MouseRelease

The event value for event 1-3 is the hot spot's tool tip or – if defined – the hot spot node's "id" attribute.
The event value for event 4 depends on the hot spot action and can be:
- the value of the "URL" attribute
- the area to zoom to
- the "cfg" parameter of a new configuration file to load
- the value of the "SelectImage" attribute

Library	
Description	Default fill opacity for the HotSpot areas
Syntax	Number
Default	25
Context	Plug-in attribute or <HotSpots> group tag
Version	5.0.4

Defines the default fill opacity for HotSpots that do not contain the "FillAlpha" attribute where "0" is transparent and "100" fully opaque.

Defining Default Attributes

The following attributes define default attributes to all HotSpot that do not contain a corresponding attribute itself.
E.g. defining a DefaultTarget of "_self" will open all hyperlinks in the frame the viewer resides in, provided that there is no individual Target attribute defined in a hotspot node.

DefaultTarget	
Description	Default target for hyperlinks
Syntax	String
Default	_blank
Context	Plug-in attribute or <HotSpots> group tag

Defines the default target(frame) for HotSpot hyperlinks.
Valid parameters are "_blank", "_self", "_top", "_parent" and names of frames inside a HTML frameset.

DefaultMove	
Description	Zoom to HotSpot area by default
Syntax	Bool
Default	true
Context	Plug-in attribute or <HotSpots> group tag

Defines the default value for all subsequent HotSpots that do not contain the "Move" attribute.

DefaultSkew	
Description	Permit skewing of HotSpots
Syntax	Bool
Default	false
Context	Plug-in attribute or <HotSpots> tag

Defines the default value for all subsequent HotSpots that do not contain the "Skew" attribute.

DefaultColor	
Description	Default color of HotSpot borders
Syntax	HexColor
Default	FF0000
Context	Plug-in attribute or <HotSpots> tag

Defines the default color of HotSpot borders.

DefaultFill	
Description	Fill the HotSpot area by default
Syntax	Boolean
Default	true
Context	Plug-in attribute or <HotSpots> group tag

Defines the default value for all subsequent HotSpots that do not contain the "Fill" attribute.
If this value is set to false, all HotSpots will be shown as outlines by default.

DefaultFillColor	
Description	Default fill color for the HotSpot areas
Syntax	HexColor
Default	FFFFFF (white)
Context	Plug-in attribute or <HotSpots> group tag

Defines the default fill color for HotSpots that do not contain the "FillColor" attribute.

DefaultFillAlpha	
Description	Default fill opacity for the HotSpot areas
Syntax	Number
Default	25
Context	Plug-in attribute or <HotSpots> group tag

Defines the default fill opacity for HotSpots that do not contain the "FillAlpha" attribute where "0" is transparent and "100" fully opaque.

Defining Individual HotSpots

Each HotSpot definition must at least contain the name of the HotSpot shape and the "spot attribute which defines the position of the HotSpot.
The tag name of each definition specifies the shape of the HotSpot.

Available default shapes include:
- rect (Square / Rectangle)
- circle (Circle / Ellipse)
- star (Star)
- polygon (Polygon of n straight lines)

HotSpot definition

```
<HotSpots>

<circle spot="1, 1, 0.35842, 0.15383, 0.61261, 0.40802, 0" tip="circle />
<rect spot="1, 10, 0.54314, 0.68446, 0.74036, 0.88168, -56" tip="rectangle" url="rect.
htm" />
<polygon coords="0.5,0 ,1,0 ,1,0.5, 0.5,0.5" ><text>some <b>formatted</b> text</text>
</polygon>

</HotSpots>
```

Please refer to the chapter *HotSpot Authoring* for useful tips on how to create HotSpots.

Please refer to the *FSI Contrib package* for a sample and instructions how to add custom link shapes. The contrib package contains sample HotSpot XML definitions as well.

HotSpot Parameters

The following attributes can be assigned to each HotSpot:

Spot	
Description	Position of the HotSpot
Syntax	String
Default	—
Context	HotSpot tag

Defines the position of the HotSpot.
It is recommended to use the plug-in *SelectFrame* to obtain this value. Please refer to the chapter "HotSpot Authoring tips" for more information on this parameter.

Coords	
Description	Coordinates for **polygon** shapes
Syntax	String
Default	—
Context	HotSpot tag

Defines the coordinates of a polygon shape. In contrast to predefined shapes, polygons do not require the "spot" parameter. Instead you need to provide the x and y coordinates (0..1) for each point determining the polygon shape. You might want to add the spot parameter to specify the scene and sceneset the polygon applies to with 3D presentations.

View	
Description	Image section to display
Syntax	String
Default	Value of spot parameter
Context	HotSpot tag
Version	3.2.3 and above

Defines the position to display when move is set to true.
If this parameter is undefined, the value of spot will be used instead.
Please refer to the chapter "HotSpot Authoring tips" for more information on this parameter.

Tip	
Description	Tooltip
Syntax	String
Default	—
Context	HotSpot tag

Defines a string to be used for a tooltip when the user moves the cursor over the HotSpot.
For long texts or *HTML formatted* texts you can add a <text> child node to each hotspot node instead of using the "Tip" parameter.

HotSpot text definition

```
<HotSpots>
...
       <Circle spot="1,1,0,0,0.6,0.4,0" >
              <text><b>Headline</b><br/>Some text</text>
       </Circle>
...
</HotSpots>
```

Move	
Description	Zoom to HotSpot area on click
Syntax	Boolean
Default	false
Context	HotSpot tag

Zoom to the image section covered by the HotSpot, if the user clicks the HotSpot.
You can use the "DefaultMove" parameter to change the default behavior.

Skew	
Description	Allow skewing of the HotSpot area
Syntax	Boolean
Default	false
Context	HotSpot tag

If this parameter is set to false the width of the hotspot area will equal the height of the HotSpot regardless of the values provided with the "Spot" parameter.
You might want to use this parameter to ensure that e.g. HotSpots defined with the <circle> tag look like circles instead of ellipses.
You can use the "DefaultSkew" parameter to change the default behavior.

Color	
Description	Color of the HotSpot border
Syntax	HexColor
Default	FF0000
Context	HotSpot tag

Defines the color of the HotSpot border.
The value has to be a 6 digit hexadecimal number like "FFFF00" for yellow or "0000FF" for blue. You can use the "DefaultColor" parameter to change the default border color.

Fill	
Description	Fill the HotSpot area
Syntax	Bool
Default	false
Context	HotSpot tag

If this value is set to false, the HotSpot will by shown as an outline.
You can use the "DefaultFill" parameter to change the default behavior.

URL	
Description	Hyperlink of the HotSpot
Syntax	URL
Default	—
Context	HotSpot tag

Defines a hyperlink to be opened if the user clicks the HotSpot.
The URL can be given relatively or absolutely.
The Target parameter (or respectively the "DefaultTarget" parameter) defines the targetframe for the hyperlink. Please refer to the "BaseURL" parameter as well.

Target	
Description	Target frame for hyperlinks
Syntax	String
Default	_blank
Context	HotSpot tag

Defines the targetframe for HotSpot-hyperlinks.
Valid parameters include "_blank", "_self", "_top", "_parent" and names of frames inside a HTML frameset.

Please refer to the "DefaultTarget" parameter for details on how to set a default target for hyperlinks.

CFG	
Description	FSI parameters to be used on click
Syntax	String (query)
Default	—
Context	HotSpot tag

If you define a CFG parameter for a HotSpot the viewer will be reinitialized with the given configuration if the user clicks the HotSpot.
This can be useful to switch from one image to another if the user clicks a HotSpot.

The syntax of the CFG parameter has to consist of FSI Parameters concatenated with the ampersand character (like a HTTP query).

Parameter1=*value1*&*Parameter2*=*value2…*

Example:
`cfg="cfg=image2&NoNav=1"`

Defining the parameter like this, the viewer would reinitialize using the configuration file "image2.fsi" providing the FSI parameter "NoNav" with a value of "true".

Please refer to the FSI Viewer manual for a list of all FSI Parameters and for details on how to create a query string that provides FSI parameters.

SelectImage	
Description	Select another image (FSI Showcase only)
Syntax	Number
Default	—
Context	HotSpot tag

Clicking the HotSpot selects another image inside FSI Showcase.
The index can be specified absolutely by providing a number only. The first image is being represented by "0".

Alternatively you can select an image relatively from the currently selected image. To select an image relatively you have to provide a number prefixed by "+" or "-", e.g. "+1" selects the next image.

Hotspot Authoring

The following section provides tips on how to obtain the string value required to define the "Spot" and the optional "View" parameter of a HotSpot.

The Spot parameter consists of 7 numbers concatenated with commas and provides the scene set, the scene, the image section (as a rectangle) and the rotation.

Example: 1, 10, 0.19056, 0.33579, 0.44862, 0.59385, 35

scene set 1, scene 10, rectangle [0.19056, 0.33579, 0.44862, 0.59385],
rotation 35 °

Although you can manually enter the parameters, it is recommended to use the plug-in *SelectFrame* to obtain this value.

Obtaining Hot Spot positions

Include the plug-in "SelectFrame" by adding the following parameters to the "_default.fsi" or any other configuration file.

Including the SelectFrame plug-in

```
<Plugins>

      <Plugin src="SelectFrame" visible="1" window="1" rotate="1" AspectRatio="1:1" />

</Plugins>
```

After including the plug-in you will be able to author
hotspots directly in FSI Viewer.

- You can use the value displayed in the "Selection" window for the "Spot" parameter of the hotspot.
- You can use the value displayed in the "View window for the optional "View" parameter.

When modifying the select box please use:

SHIFT	to lock the aspect ratio of the selection
CTRL	to resize the selection relative to the center
ALT	to rotate the selection
ALT + SHIFT	to rotate the selection in steps of 5 degrees

Defining Polygon HotSpots

Besides using predefined shapes as hotspots you can additionally add polygons. In this case you need to specify the X and Y coordinates for each point of the polygon using the "coords" attribute of the hotspot node.

HotSpot Polygon definition

```
<HotSpots>

      <Polygon spot="1,1" coords="0.5,0 ,1,0 ,1,0.5, 0.5,0.5" />

</HotSpots>
```

Please note that the "spot" attribute is required only, if you want to specify the target scene and/or scene set for the hotspot when using a 3D image.

The "coords" attribute contains [0..1] floating point coordinates where 0 is left or top and 1 is right or bottom. The coordinates for the polygon points defining the polygon shape need to be specified in the form:

x1,y1, x2,y2, x3,y3 …

Theoretically you can use any number of coordinates (points), but keep in mind that more coordinates and more polygons require more CPU power on the user‘s system. Using very complex polygons with hundreds of coordinates will lead to poor performance as the polygons will be drawn by FSI Viewer at runtime.

HotSpot Z-Order

When defining HotSpots that overlap each other, you have to keep the z-order of the HotSpots in mind. The z-order of HotSpots is determined by the sequence in which they are defined in the configuration file, from bottom to top.

As a rule of thumb you should define big Hot Spot areas first to ensure that smaller Hot Spots are not completely hidden beneath.

Plug-in JavaScript Bridge

Plug-in Target:
Scripting

Plug-in Location:
/plugins/jsbridge.plg

Function:
Integrates a JavaScript interface to enable interaction between FSI Viewer and JavaScript contained in the hosting HTML page.

Syntax:
`<Plugin src="jsbridge" />`

Requirements:
Browser supporting live connect between Flash and JavaScript (Livewire)

This plug-in can be optionally obtained.
Using the plug-in without a valid licence key activates the evaluation mode.

The "JavaScript Bridge" plug-in provides a scripting interface to control FSI Viewer using JavaScript. Using the JavaScript bridge you can control viewer functions like "ZoomIn" and "Reset", you can remotely press/release Buttons inside the FSI Viewer and you can zoom to specified image sections directly. Additionally you can implement a JavaScript function receiving FSI event notifications.

Though the integration of the plug-in is very easy, basic knowledge of JavaScript is required to use JSBridge plug-in – especially regarding browser dependent JavaScript implementations.

Implementing the JavaScript Bridge plug-in

1) Add an ID-attribute to your FSI Viewer Object and Embed tag

```
<object id="fsiviewer" classid= ...>
<embed NAME="fsiviewer"...>
```

This way you are able to access the viewer object via Javascript.

2) Provide the appropriate action via SetVariable command
To control FSI Viewer from JavaScript you have to pass the appropriate command by setting the "`FSICMD` " variable of the FSI Viewer movie clip using the SetVariable() function.

Example: Button "Zoom In" (MSIE)

```
<input type="button" value="Zoom In"
       onMouseDown="document.all.fsiviewer.SetVariable('FSICMD', 'ZoomIn');"
       onMouseUp  ="document.all.fsiviewer.SetVariable('FSICMD', ''); >
```

Command Parameters

There are 3 different command types you can pass to FSI Viewer, FSI Pages or FSI Showcase:

1. Initiate an action directly, e.g. "`ZoomIn`" or "`Reset`"
2. Press and release of FSI Viewer buttons, e.g. "`Button:Mouse_Mode_0`"
3. Zoom to a specified image section, e.g. "`Goto:1,6,0.4,0.2,0.6,0.4`"
   Please see the "HotSpot Authoring" section for details on how to obtain valid image section parameters.

The following table lists possible values you can pass to FSI Viewer.
Please note that especially regarding the "Button:" command there might be more or not all commands listed available, depending on your FSI Viewer skin and integrated plug-ins.

All command strings have to be provided exactly as stated in the lists below, especially regarding upper-/ lowercase writing and white spaces.

1) Direct Commands

Usage: `FSIViewer.SetVariable('FSICMD', '[command]');`

To stop the initiated action use: `FSIViewer.SetVariable('FSICMD', ',);`

Command	Description
" (empty string)	Stop current action / Release Button
Reset	Reset Viewer
ZoomIn	Start zooming in
ZoomOut	Start zooming out
Up	Start panning upwards
Down	Start panning downwards
Left	Start panning to the left
Right	Start panning to the right
RotateLeft	Start rotating counter-clockwise (Z-axis)
RotateRight	Start rotating clockwise (Z-axis)
NextScene	Start rotating clockwise (Y-axis)
PreviousScene	Start rotating counter-clockwise (Y-axis)
NextSceneSet	Start rotating clockwise (X-axis)
PreviousSceneSet	Start rotating counter-clockwise (X-axis)
__DebugWindow	Show / Hide the debug window (if available)
__InfoWindow	Show / Hide the "about"-window (if available)
FSI Showcase only:	**Description**
PreviousImage	Select previous Image
NextImage	Select next Image
SelectImage	Select image *n* **Important:** Use SetVariable("newImageIndex", *n*) where *n* is [0 .. images] prior to sending this command Alternatively you can select an image relatively from the currently selected image. To select an image relatively you have to set "newImageIndex" to a number prefixed by "+" or "-", e.g. "+1" selects the next image.
SwitchThumbs	Show / Hide Thumb bar

FSI Pages only:	Description
FirstPage	Go to first page.
LastPage	Go to last page.
NextPage	Go to next page.
PreviousPage	Go to previous page.
GotoPage	Go to page *n* **Important:** Use SetVariable("newImageIndex", *n*) where *n* is [0 .. pages] prior to sending this command Alternatively you can specify a page relative to the current page. To go to a page relatively you have to set "newImageIndex" to a number prefixed by "+" or "-", e.g. "+1" is the same as "NextPage".
Print	Open or close the print dialog.
Save	Open or close the save dialog.
Search	Open or close the search dialog.
ZoomLeftPage	Zoom the left page
ZoomRightPage	Zoom the right page
NormalPageView	Exit zoom view
ShowHidePageIndex	Show or hide the page index (thumbnail view)

2) Button press / release

Using this command remotely presses/releases a button in FSI Viewer / FSI Showcase. There can not be more than one button pressed at a time. Pressing another button (by script) will automatically release the previous button.

Usage:
```
FSIViewer.SetVariable('FSICMD', 'Button:[ButtonID]');
```

To release the button previously pressed use:
```
FSIViewer.SetVariable('FSICMD', ',);
```

ButtonID	Description
'' (empty string)	Release previously pressed button
Mouse_Mode_*n*	Activate mouse mode *n* . 0 – Zoom 1 – Pan 2 – Rotate 3D 3 – Rotate (Z-axis) There might be more mouse modes available depending on loaded plug-ins
Tool_Reset	"Reset" button, see command "Reset"
Tool_ZoomIn	"Zoom In" button, see command "ZoomIn"
Tool_ZoomOut	"Zoom Out" button, see command "ZoomOut"
Tool_SwitchUI	Show / Hide FSI Viewer user interface
Plugin_SwitchMusic	Toggle music on or off
Plugin_ExtendedNavigation	Toggle display of ExtendedNavigation plug-in
Plugin_HistoryBack	Click "History Back" button of the history plug-in
Plugin_HistoryForward	Click "History Forward" button of the history plug-in

Identifying commands of buttons not listed in the table

There might be some button identifiers that are not listed below, especially buttons integrated by other plug-ins. To obtain command identifiers of these buttons follow the steps below:

1. Activate the debug window of the FSI viewer (set parameter "debug" to "true").
2. Set the explore parameter of the jsbridge plug-in to "true".
3. Click the desired button inside FSI Viewer and copy the identifier from the debug window.

3) Navigating to image sections

To navigate to a specific image section you simply use the "Goto" command followed by a valid image section parameter. Please refer to the chapter "*Debug Mode*" for details on how to obtain image section parameters.

Example: Button "Zoom to image section" (MSIE)

```
<input type="button" value="Zoom to section"
      onClick="document.all.fsiviewer.SetVariable('FSICMD', 'Goto:1,3,0,0,0');">
```

Implementing JavaScript callbacks

Using JavaScript callbacks enables you to react to FSI Viewer, FSI Pages or FSI Showcase events by executing JavaScript code. The JSBridge plug-in calls a JavaScript function using the Flash FSCommand architecture providing an event identifier and a parameter.

Please follow the steps below to implement JavaScript callbacks from FSI Viewer:

Step 1. Enable JavaScript callbacks by setting the plug-in parameter "Callback" to "true", e.g.

Example: Enabling JavaScript Callbacks

```
<plugin src="jsbridge" callback="true" />

Or alternatively via HTTP query:

...fsi.swf?cfg=foo&jsbridge_callback=1
```

Step 2. Add a unique identifier to the <object> and the <embed> tag of FSI Viewer / FSI Showcase.

Example: Adding ID / NAME parameters

```
<object id="fsiviewer" classid= ...>
<Param name="movie" value="...>
...
<embed NAME="fsiviewer"...>
...
</object>
```

Step 3. Add a JavaScript function to your HTML page receiving the event notifications via FSCommand like in the example below.

IMPORTANT

The function name has to be exactly like in the example below and it has to start with the ID/NAME parameter specified in step 2.

If you do not use the ID/NAME "fsiviewer" you have to replace the corresponding script sections in the example below (printed bold) by your ID/NAME parameter.

Example: JavaScript Callback Function

```
<script language="javascript" type="text/javascript">

function fsiviewer_DoFSCommand(fsi_event, params) {
       switch(fsi_event){
              case "LoadProgress":
              ...
              break;
       }
}

// Hook for Internet Explorer
if (navigator.appName && navigator.appName.indexOf("Microsoft") != -1 &&
navigator.userAgent.indexOf("Windows") != -1 && navigator.userAgent.indexOf("Windows
3.1") == -1) {
       document.write('<SCRIPT LANGUAGE=VBScript\> \n');
       document.write('on error resume next \n');
       document.write('Sub fsiviewer_FSCommand(ByVal fsi_event, ByVal params)\n');
       document.write('  call fsiviewer_DoFSCommand(fsi_event, params)\n');
       document.write('end sub\n');
       document.write('</SCRIPT\> \n');
}
</script>
```

After following the 3 steps described before you will receive the following event notifications:

Event (fsi_event)	Parameter (params)	Description
InitComplete	—	The viewer has finished parsing the configuration and is about to start loading image data.
ImageInfo	String	Provides concatenated image information. ***See details below this table.**
LoadingComplete	—	The initial load progress of an image completed.
LoadProgress	Number (percent)	Number corresponding to the progress bar inside FSI Viewer.
Press	Button ID	The button "ID" has been pressed.
Release	Button ID	The button "ID" has been released.
ToolTip	String (escaped)	The tool tip "String" is about to be displayed. ***See details below this table.**
View	Image Section	The image section currently displayed. Can be used with the "Goto" command.
ImageUrl	URL template	Provides a template that can be used to build an image URL corresponding to the image being displayed in FSI Viewer. ***See details below this table.**
Action	Action ID	The action "ID" is currently being executed.
Modal	true / false	The viewer enters a passive (true) or an active (false) state.
MouseMode	Mouse Mode ID	The mouse mode "ID" has been activated.
Zoom	Magnification (percent)	The magnification has changed.
ImageSelected	Image index (0..nImages)	A new image has been selected. (**FSI Showcase only**)
FSI Pages only		
onPagesInitComplete	Integer	Called after the FSI Pages Add-on is ready for use. The parameter provides the total number of pages.
onPagesGotoPage	Integer	Called each time before the user goes to a different page. The parameter provides the target page number.
onPagesPageChanged	Integer	Called each time after the current page has changed. The parameter provides the index of the page currently visible.
onPagesProgress	Integer	Called each time the load progress of FSI Pages changes. The parameter provides the number of pages remaining in the load queue.
onPagesShowIndex	Boolean	Called each time the user shows (true) or hides (false) the page index.
onPagesShowZoom	Boolean	Called each time the user starts (true) or ends (false) page zoom.
onPagesLinkClick	URL	Called each time the user clicks a link. The parameter contains the URL.

Please note that some plug-ins provide additional events. Please refer to the "events" parameter of the corresponding plug-in for details.

The recommended way to react to an event is to extend the switch...case block of the `fsiviewer_DoFSCommand(fsi_event, params)` callback function introduced in Step 3.

Detailed Event Parameter Description

INFO

The "Info" event parameter provides attributes of the source image currently being displayed.

Example:
`width=8000;height=6000;TilesX=4;TilesY=3;SceneSets=;ImageIndex=4`

The attribute "ImageIndex" applies to FSI Showcase or FSI Pages only and provides the 0-based index of the image currently selected.

ImageURL

To receive this notification you have to set the "ImageURLs" parameter of the plug-in to "true".
(Please refer to "Plug-in Parameters" below for details)

The "ImageURL" event parameter provides an URL template that can be used to retrieve the image currently displayed in FSI Viewer.

Example:
`http://imageserver.domain/fsi/server?source=images/zoomimage.fpx&width=[width]&height=[height]&left=0&top=0&right=1&bottom=1&template=[template]`

The URL template can for example be used to set the "src" attribute of an <IMG> tag.

Prior to using the template you have to replace the place holders (bold, in brackets) with valid values.

The **[template]** place holder applies to FSI Servers only.

ToolTip

The "ToolTip" event parameter provides the tool tip text that is about to be displayed.
The values depend on the tool tip language selected.
You have to use the JavaScript function **decodeURIComponent ()** before displaying this parameter.

JSBridge Plug-in Parameters

AllowDomains	
Description:	Sub(domains) allowed to acces the JSBridge interface
Syntax:	Domain list separated by ","
Default:	Subdomain of FSI Viewer, "www.domain.tld", "domain.tld","*.domain.tld"
Context:	Plug-in attribute
Version:	4.1.0 or above

With Adobe Flash version 8 an additional limitation regarding Javascript to Flash access has been introduced. Starting with version 8, only a script from the same subdomain as the Flash movie (e.g. "fsi.domain.com") is allowed to access Flash / the JSBridge interface.

With FSI Viewer and FSI server it is very likely that you use FSI Viewer from "fsi.yourdomain.tld" while your HTML document resides in the subdomain "www.yourdomain.tld". Therefore JSBridge plug-in allows script access from the subdomain "www.yourdomain.tld" and from the main domain "yourdomain.tld" by default.

In case you need to access JSBridge from other (sub)domains you can specify a list of subdomains separated by "," using this parameter, e.g.:
<allowdomains value="www.domain.com,www.domain2.com" />

Explore	
Description:	Activate debug output of Button IDs
Syntax:	Boolean
Default:	false
Context:	Plug-in attribute

If enabled the debug window will display the Button ID of each button pressed inside FSI Viewer. This way you are able to determine Button IDs not listed in this documentation.
Additionally the debug window will output each command received by the JSBridge plug-in.

Callback	
Description:	Enable JavaScript callbacks
Syntax:	Boolean
Default:	false
Context:	Plug-in attribute

If enabled the events of FSI Viewer, FSI Pages or FSI Showcase will be forwarded to a specific JavaScript function. Please refer to the section "*Implementing JavaScript Callbacks*" for details

ImageUrls	
Description:	Provide ImageUrl templates with JavaScript callbacks
Syntax:	Boolean
Default:	false
Context:	Plug-in attribute

This parameter depends on the "Callback" parameter above. If "Callback" and "ImageUrls" parameter have been set to "true", "ImageUrl" notifications providing image URL templates will be sent to the JavaScript callback function.
Please refer to the "ImageURL" callback identifier for details.

Plug-in LargeToolTips

Plug-in Target:
User interface of FSI Pages

Plug-in Location:
/plugins/largetooltips.plg

Function:
Display multi-line tool tips with HTML formatted text.

Syntax:
```
<Plugin src="largetooltips"/>
```

This plug-in adds multi-line tool tips displaying HTML formatted text to links in FSI Pages add-on. The following HTML tags can be used in the tool tip text data:

```
<b>…</b> <i>…</i> <u>…</u>
<br/>
<font size="n" color="#RRGGBB" >…</font>
<p align="…">…</p>
<img src="..." width="..." height="...">
```

Note: Certain limitations apply when using the img-tag: The width and height attributes are mandatory and the URL in the src-attribute must end width .jpg or .png. If you are using a single source imaging server to provide these images, then consider appending something like &ext=.jpg to the src-URL. Furthermore images will always be inserted into the textflow, if it is desired to continue the text below the inserted image, extra line breaks will be needed.

Plug-in Parameters

Width	
Description	Width of the tool tip box
Syntax	Pixels
Default	200
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Width of the tool tip box. The height of the box will be adjusted according to the length of the contained text.

Delay	
Description	Delay before displaying a tool tip
Syntax	Number (milliseconds)
Default	600
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

You can specify a delay in milliseconds (1000 milliseconds = 1 sec) after pointing at a link area and before displaying a tool tip. Use 0 to display the tool tips immediately without a delay.

Shadow	
Description	Display a drop shadow
Syntax	Boolean
Default	true
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Display (true) or hide (false) a drop shadow for the tool tip box.

BackgroundColor	
Description	Background color of the tool tip box
Syntax	HexColor
Default	FFFFFF
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Background color of the tool tip box.

BackgroundAlpha	
Description	Opacity of the background of the tool tip box
Syntax	Number
Default	90
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Defines the opacity of the tool tip background from 0 (invisible) to 100 (opaque).

HighlightColor	
Description	Border highlight color
Syntax	HexColor
Default	999999
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

6-digit hexadecimal color value specifying the highlighted (top-left) part of the tip box border.

ShadowColor	
Description	Border shadow color
Syntax	HexColor
Default	999999
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

6-digit hexadecimal color value specifying the shadow (bottom-right) part of the tip box border.

LineWidth	
Description	Thickness of the tip box border
Syntax	Number
Default	2
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Thickness of the tip box border in pixels.

TextFont	
Description	Defines the font of the tool tip text
Syntax	String
Context	Plug-in node or HTTP query prefixed with "largetooltips_"
Version	5.0.0

Defines the font of the tool tip text. Please note that the desired font must be available on the users machine.
You need to specify the font name (e.g. "Courier New" or use one of the Flash Player predefinitions "_sans", "_serif" or "_typewriter")

TextColor	
Description	Default text color
Syntax	HexColor
Default	000000
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Default text color of text in the tip box. To use different colors in a tool tip you can use the HTML tag <font color="RRGGBB"/> in the tool tip text data.

TextSize	
Description	Default text size
Syntax	Number
Default	11
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Default text size of text in the tip box. To use different sizes in a tool tip you can use the HTML tag <font size="n"/> in the tool tip text data.

CornerRadius	
Description	Radius of box corners
Syntax	Pixel
Default	20
Context	Plug-in node or HTTP query prefixed with "largetooltips_"

Radius of the box corners. Use "0" for straight edges and positive values for rounded edges.

Plug-in Magnifier

Plug-in Target:
User interface

Plug-in Location:
/plugins/magnifier.plg

Function:
Adds a magnifier to the user interface

Syntax:

```
<Plugin src="magnifier" visible="true" size="10" />
```

The "Magnifier" plug-in adds a magnifier to the user interface that can be dragged using the mouse. Using the magnifier plug-in does not cause additional data traffic, as the plug-in enlarges already loaded image data by software.
The magnifier can be hidden and shown using a corresponding button in the menu bar. The level of magnification and the magnifier size can be defined by plug-in parameters.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attribute

Specifies the space in pixels left of the menu button of the plug-in.

Visible	
Description	Initial state of visibility
Syntax	Boolean
Default	false
Context	Plug-in attribute

Defines the initial visibility of the magnifier.

Size	
Description	Size of magnifier
Syntax	Number
Default	10
Context	Plug-in attribute

Defines the overall size of the magnifier. Possible values are between 4 and 16.
Please note that greater values require more CPU power.

Magnification	
Description	Level of magnification
Syntax	Number
Default	2
Context	Plug-in attribute

Defines the level of magnification where 1 means no magnification, 2 means a magnification of 200% and so on. The maximum level of magnification is 8.

Reflections	
Description	Show or hide glass reflections
Syntax	Bool
Default	true
Context	Plug-in attribute

Show (true) or hide (false) glass reflections of the magnifier.

Handle	
Description	Show or hide magnifier handle
Syntax	Bool
Default	true
Context	Plug-in attribute

Show (true) or hide the handle of the magnifier glass.

HandleSize	
Description	Size of the handle in percent
Syntax	Number
Default	100
Context	Plug-in attribute

Size of the magnifier handle.

Plug-in MaxZoom

Plug-in Target:
User interface

Plug-in Location:
/plugins/maxzoom.plg

Function:
Zoom to physical resolution of the image.

Syntax:
```
<Plugin src="maxzoom" />
```

The "MaxZoom" plug-in expands the user interface by a button to quickly zoom to the physical resolution of the source image.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attribute

Specifies the space in pixels left of the menu button of the plug-in.

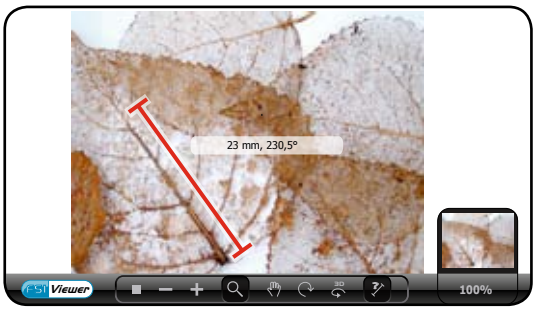
Plug-in Measure

Plug-in Target:
User interface

Plug-in Location:
/plugins/measure.plg

Function:
Provides distance and angle measuring

Syntax:
```
<Plugin src="measure" />
```



This plug-in can be optionally obtained.
Using the plug-in without a valid licence key activates the evaluation mode.

The Measure plug-in provides an additional mouse mode which enables the user to measure distances and angles inside the FSI Viewer by clicking & dragging.

Pressing SHIFT locks the angle to 45 degree steps.
Pressing CTRL moves the measurement line.

Mouse Mode

The mouse mode id for the measuring mode is 100.
You can use this id to set the InitialMouseMode parameter of the FSI Viewer.
e.g.
```
<InitialMouseMode value="100" />
```
or
```
?cfg=foo&InitialMouseMode=100
```

Setting up the Plug-in

The Measure Plug-in requires the real width of the entire image to enable distance measuring.

Example 1:
If you already know the entire width of the image simply add the "ImageWidth" and the "Suffix" parameter to the plug-in tag:

If the width of the entire image is 120.5 inches the corresponding plug-in tag looks as follows:

```
<plugin src="measure" ImageWidth="120.5" Suffix=" inches" />
```

Please note:
You might as well define all plug-in parameters via query or inside the <options> group by using the prefix "measure_".

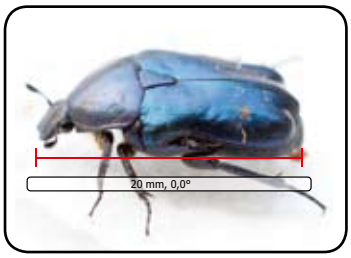
E.g.:

```
?cfg=image&Measure_ImageWidth=120.5&Measure_Suffix=%20inches
```

Example 2:
If you know the width of a part of the image, but you do not know the width of the entire image, you should follow the steps below:

1. Add the measure plug-in without any parameters.
   <Plugin src="measure" />
2. Open the image inside the FSI Viewer and choose the "Measure" mouse mode.
3. Measure the part of the image you know the real width of.
   (The Measure plug-in will display a decimal number *n* between 0 and 1)
4. The value *w* for the ImageWidth parameter is:

   w = RealWidth / n.
   The Measure plug-in displays *n=20*.
   The real length *L* of the bug is 53 mm.
   ImageWidth *w = L / n = 50 / 20 = 2,65*



The corresponding plug-in tag is the following:

```
<plugin src="measure" ImageWidth="2.65" Suffix=" mm" />
```

Please note:
You might as well define all plug-in parameters via query or inside the <options> group by using the prefix "`measure_`".

E.g.:

```
?cfg=engine&Measure_ImageWidth=2.65&Measure_Suffix=%20mm
```

Accuracy

The plug-in does not support any perspective correction.
This means that measuring scanned maps, diagrams will be accurate, measuring 3 dimensional objects is less accurate.

Plug-in Parameters

The following attributes can be assigned to the <plugin> tag.
You might as well define all plug-in parameters via query or inside the <options> group by using the prefix "`measure_`" with the parameter name, e.g. "`measure_ImageWidth`"

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attribute

Specifies the space in pixels left of the menu button of the plug-in.

ImageWidth	
Description	Real width of the entire image
Synta	Floating Point Value
Default	100.0
Context	Plug-in attribute

Defines the real width of the entire image (e.g. 120 inches).
Please see the explanation in the previous section.

Prefix	
Description	Defines a prefix for the length value
Syntax	String
Default	---
Context	Plug-in attribute

Defines the prefix for the length value e.g. "length: ".

Suffix	
Description	Defines a suffix for the length value
Syntax	String
Default	mm
Context	Plug-in attribute

Defines the suffix for the length value e.g. "inches". Usually this is a length unit.

Decimals	
Description	Number of decimals of the length value
Syntax	Number
Default	1
Context	Plug-in attribute

Defines the number of decimals of the length value.

ShowText	
Description	Display the current measurement string
Syntax	Boolean
Default	true
Context	Plug-in attribute

Hides the text displaying the current measurement if set to "false".

ShowAngle	
Description	Display current angle
Syntax	Boolean
Default	true
Context	Plug-in attribute

If set to true the plug-in appends a text representing the current angle in degrees to the text display. E.g. "120.2 inches, 43.2°".

LineColor	
Description	Defines the color of the measure lines
Syntax	HexColor
Default	FF0000
Context	Plug-in attribute

6-digit hexadecimal color value defining the color of the measure lines.

TextColor	
Description	Defines the color of the text
Syntax	HexColor
Default	000000
Context	Plug-in attribute

6-digit hexadecimal color value defining the color of the text.

TextBorder	
Description	Defines the color of the text border
Syntax	HexColor or "false"
Default	000000
Context	Plug-in attribute
Version	4.1.0 or above

6-digit hexadecimal color value defining the color of the text border.
Use "false" for no border.

TextSize	
Description	Size of the text
Syntax	Number (pixel)
Default	12
Context	Plug-in attribute

Text size for the textual output in pixels.

BGColor	
Description	Defines the color of the text background
Syntax	HexColor
Default	FFFFFF
Context	Plug-in attribute

6-digit hexadecimal color value defining the background color of the text.

BGAlpha	
Description	Defines the opacity of the text background
Syntax	Number
Default	80
Context	Plug-in attribute

Defines the opacity of the text background (0...100).

GlowRange	
Description	Size of the glow effect
Syntax	Number
Default	4
Context	Plug-in attribute

The glow range is a highlighted area surrounding the measure tool to improve it's visibility. Using the GlowRange parameter you can define the size of the "glowing" area. Use "0" to disable the glow range.

GlowAlpha	
Description	Opacity of the glow range
Syntax	Number
Default	25
Context	Plug-in attribute

Opacity of the glow range surrounding the measure tool.

GlowColor	
Description	Color of the glow range
Syntax	HexColor
Default	FFFFFF
Context	Plug-in attribute

6-digit color defining the color of the glow range surrounding the measure tool.

Events	
Description	Send plug-in event notifications
Syntax	Boolean
Default	false
Context	Plug-in attribute
Version	3.2.5 and above

Specifies whether the plug-in sends plug-in event notivications. Plug-in events can be handled when using JSBridge plug-in or FSI Viewer Component.

The following events apply to this plug-in:

Event ID	Parameter	Description
onMeasureEnable	Boolean	The measure tool gets enabled or disabled.
onMeasureStart	String	The user starts measuring. The parameter provides the current distance and angle as displayed in the measure tool.
onMeasure	String	The user drags the measure tool. The parameter provides the current distance and angle as displayed in the measure tool.
onMeasureStop	String	The user stopped measuring. The parameter provides the current distance and angle as displayed in the measure tool.

Plug-in Mousemodes

Plug-in Target:
User interface

Plug-in Location:
/plugins/mousemodes.plg

Function:
Integrates buttons into the menu bar to select the mouse mode.

Syntax:
```
<Plugin src="mousemodes" />
```

The "Mousemodes" plug-in provides 3 buttons (2D images) or 4 buttons (3D presentations) to select the current mouse mode.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu buttons
Syntax	Number
Default	0
Context	Plug-in Tag

Specifies the space in pixels left of the menu button(s) of the plug-in.

Mode *n*	
Description	Removes the button for mouse mode *n* from the menu bar
Syntax	Boolean
Default	—
Context	Plug-in Tag

Provides the possibility to hide specific mouse mode buttons.

Remove Drag-Button
`<Plugin src="mousemodes" Model="false" />`

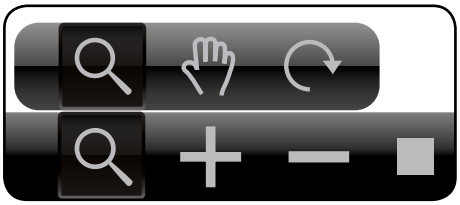
The example above shows how to hide the button for mouse mode 1 (Drag).

Mode n	Mouse Mode
Mode0 	Zoom
Mode1 	Pan
Mode2 	Rotate 3D (X/Y Axis)
Mode3 	Rotate 2D (Z Axis)

Sequence	
Description	Sequence of the buttons
Syntax	String
Default	0,1,3,2
Context	Plug-in Tag
Version	3.5.0 or higher

You can alter the sequence of the buttons by providing the modes separated by commas (see the table above).
Example: "1,0,2,3" alters the sequence of the buttons to
"Pan, Zoom, Rotate 3D, Rotate 2D".

Plug-in MousemodeSelect



Plug-in Target:
User interface

Plug-in Location:
/plugins/mousemodeselect.plg

Function:
Integrates a button and a dropdown menu into the menu bar to select the mouse mode.

Syntax:

```
<Plugin src="mousemodeselect" />
```

The "Mousemode Select" plug-in adds a button and a dropdown menu to the menu bar to select the mouse mode.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in Tag

Specifies the space in pixel left of the menu button of the plug-in.

Mode *n*	
Description	Removes the button for mouse mode *n* from the menu
Syntax	Boolean
Default	—
Context	Plug-in attributes

Provides the possibility to hide specific mouse mode buttons in the dropdown menu.

Remove Drag-Button
`<Plugin src="mousemodes" Model="false" />`

The example above shows how to hide the button for mouse mode 1 (Drag).

Mode n	Mouse Mode
Mode0 	Zoom
Mode1 	Pan
Mode2 	Rotate 3D (X/Y Axis)
Mode3 	Rotate 2D (Z Axis)

Plug-in Music

Plug-in Target:
User interface

Plug-in Location:
/plugins/music.plg

Function:
Adds streaming background music or sound to the FSI Viewer.

Syntax:

```
<Plugin src="music" song="music.mp3" autoplay="true" />
```

The "Music" plug-in adds a background sound or music to FSI Viewer.
Additionally a button will be added to the menu bar to switch the music on or off.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attribute

Specifies the space in pixels left of the menu button of the plug-in.

AutoPlay	
Description	Start playing the sound on startup
Syntax	Boolean or String
Default	false
Context	Plug-in attribute

Set the value to Boolean "true" to make the sound play on startup.
From version 1.3 of the plug-in you can additionally set the value to "onLoad" to start the sound after image data has been loaded completely.

Song	
Description	Path to an MPEG3 audio file (.mp3)
Syntax	URL
Default	—
Context	Plug-in attribute

Defines an absolute or relative path to an MPEG3 file (*.mp3). If you specify a relative path (instead "http://..." or "https://...") the path needs to be provided relative to the "music" subdirectory of the "/fsi" folder.

Loop	
Description	Loop / Play the sound once
Syntax	Bool
Default	false
Context	Plug-in attribute

Defines if the sound will be played once or in a continuous loop.

PauseOnStop	
Description	Pause music rather than stop when clicking the plug-in button
Syntax	Boolean
Default	false
Context	Plug-in attribute
Version:	4.0.7

By default pressing the plug-in button stops/plays the music. Setting this parameter to "true" makes the plug-in pause/resume when clicking the plug-in button so that the song does not restart from the beginning.

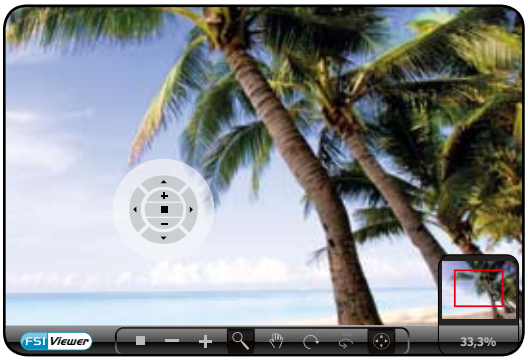
Volume	
Description	Sound volume
Syntax	Number (0-100)
Default	—
Context	Plug-in attribute

Defines the volume of the sound from 0 (quiet) to 100 (maximum).

KeepState	
Description	Keep Stop/Play state when selecting images
Syntax	Boolean
Default	true
Context	Plug-in attribute

For viewers with multiple images only.
Keep the playing state and ignore the "AutoPlay" parameter when switching from one image to another. This way "AutoPlay=true" does not start the sound if the user switched the music off.

Plug-in NavExtension



Plug-in Target:
User interface

Plug-in Location:
/plugins/navextension.plg

Function:
Additional navigation control.

Syntax:

```
<Plugin src="navextension" visible="true" />
```

The "NavExtension" plug-in adds a draggable navigation control to the FSI Viewer user interface that can be hidden and shown with a corresponding button in the menu bar. The control automatically adds buttons for scenes and scene sets depending on the image configuration.

Plug-in Parameters

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attributes

Specifies the space in pixels left of the menu button of the plug-in.

Visible	
Description	Initial display state
Syntax	Boolean
Default	false
Context	Plug-in attribute

Defines the initial visibility state of the control.

X	
Description	Initial X position in pixel
Syntax	Number
Default	0
Context	Plug-in Tag

Initial horizontal position of the control in pixels relative to the top-left corner of the viewer.

Y	
Description	Initial Y position in pixel
Syntax	Number
Default	0
Context	Plug-in Tag

Initial vertical position of the control in pixels relative to the top-left corner of the viewer.

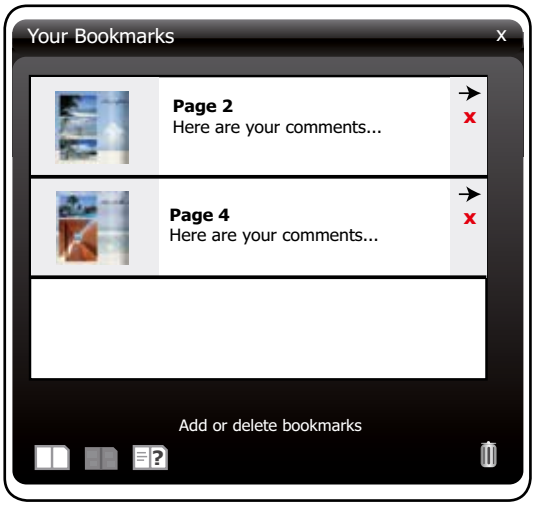
Plug-in Notepad

Plug-in Target:
User interface of FSI Pages

Plug-in Location:
/plugins/notepad.plg

Function:
Creating and displaying bookmarks with text and image.

Syntax:
```
<Plugin src="notepad" />
```



Enterprise editions of FSI Viewer contain this plug-in by default.
For all other editions this plug-in can be optionally obtained.

The notepad plug-in enables the user to create bookmarks for an image section with descriptive text. The available bookmark buttons and options depend on the Add-on you are using:

Add-on	Description
FSI Pages	- Bookmark current page(s) - Bookmark product link * - Bookmark image section
FSI Showcase	- Bookmark current image - Bookmark image section of the current image
None	- Bookmark entire image - Bookmark image section

* for bookmarks containing descriptive text (<text> nodes in XML page data, e.g. for the LargeToolTip plug-in) this text is being used as default text for the bookmark (see image above)

Unique ID

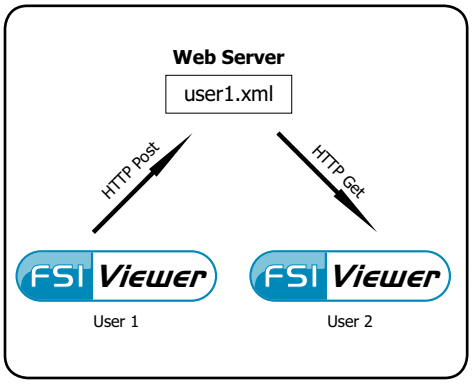
It's crucial to specify a unique ID for each application of the notepad plug-in in order to store the notepad data in a separate data set on the user's system. Otherwise the user might see bookmarks for another image or catalog.
You can either use the global FSI Viewer parameter "UniqueID" or use the Notepad plug-in parameter. Please refer to the "UniqueID" parameter description below for details.

Persistent Data Storage

The notepad uses an internal Flash technique to permanently store the notepad data on the users system. No server side script or cookies are required.

Please note that the notepad data does not store the thumbnail images, but the URL to the thumbnail instead. If you remove an image from your imaging server the user will not see the thumbnail any more when opening the notepad later on.

Posting and Loading data



The notepad plug-in provides the functionality to post bookmarks in XML format to a web server. Additionally the plug-in is capable of reading an XML data file from the server.

Using a little server side script you can this way setup a number of interesting communication systems. For example a "*send bookmarks by email*" option.

In this case you store the XML data file received from "User1's" notepad plug-in on a web server and email a link to an FSI Viewer instance including the name of the stored XML file returned by the web server to User2. This way User2 can see the bookmarks and remarks created by User1.

Additionally you can provide predefined notepad data, for example to present hot products, special offers and alike. In this case you provide the URL to a predefined XML data file using the "LoadXML" parameter. This way the notepad on the client computer loads the notepad data from this file instead of using any data stored locally on the user's computer.

Plug-in Parameters – Basic

Required Parameter !

UniqueID	
Description	Unique String used to store notepad data locally
Syntax	String (see description)
Default	---
Context	Plug-in attribute

A unique string identifying the notepad data.
Using the same (or no) UniqueID for different images or image collections (catalogs) will present false bookmarks in the notepad. You can for example use the path of the image or the name of a catalog. Do not use the following characters in UniqueIDs: [Space] ~ % & \ ; : " ' , < > ? #
Alternatively to the plug-in parameter you can use the global FSI Parameter "UniqueID" to specify a unique identifier.

Revision	
Description	Revision of the notepad data
Syntax	Number
Default	1
Context	Plug-in attribute

You can use different revisions for the same UniqueID.
Changing the revision invalidates all bookmarks for a given UniqueID.
You might for example change the revision if you change, add or remove pages in an FSI Pages catalog.

Plug-in Parameters – Layout and Appearance

Visible	
Description	Show or hide the notepad on startup
Syntax	Boolean
Default	false
Context	Plug-in attribute

Show (true) or hide (false) the notepad on startup.

Width, Height	
Description	Width and height of the notepad window
Syntax	Number (pixels)
Default	320 (width) and 272 (height)
Context	Plug-in attribute

Width and height in pixels of the notepad window.

ImageWidth, ImageHeight	
Description	Max. width and height of thumbnail images
Syntax	Number (pixels)
Default	80 (width) and 66 (height)
Context	Plug-in attribute

The maximum width and height of thumbnail images in pixels.

BackgroundColor	
Description	Max. width and height of thumbnail images
Syntax	HexColor(s)
Default	FFFFFF,DDDDDD
Context	Plug-in attribute
Version	Obsolete from version 5.0.0 (defined by FSI Viewer Skin)

Background color of the notepad window. You can specify a solid color or use multiple hexadecimal color values concatenated by, " to specify a color gradient, e.g. "FF0000,00FF00,FF0000".

BackgroundGradientAngle	
Description	Angle of the background color gradient
Syntax	Number (degrees)
Default	45
Context	Plug-in attribute
Version	Obsolete from version 5.0.0 (defined by FSI Viewer Skin)

Angle of the background color gradient in degree. The default value 45 creates a gradient from top left to bottom right. Has no effect if you used a single color value for the BackgroundColor parameter.

FrameWidth	
Description	Width of the window frame
Syntax	Floating Point Number (pixels)
Default	1.1
Context	Plug-in attribute
Version	Obsolete from version 5.0.0 (defined by FSI Viewer Skin)

Width of the window border.

FrameColor	
Description	Color of the window frame
Syntax	HexColor
Default	000000
Context	Plug-in attribute
Version	Obsolete from version 5.0.0 (defined by FSI Viewer Skin)

Color of the window border.

EdgeRadius	
Description	Radius of the window corners
Syntax	Number
Default	6
Context	Plug-in attribute
Version	Obsolete from version 5.0.0 (defined by FSI Viewer Skin)

Radius of the window corners. Use "0" for straight edges.

ImageBackgroundColor	
Description	Color of the background behind the thumbnails
Syntax	HexColor
Default	EEEEEE
Context	Plug-in attribute

Color of the background behind the thumbnails.

ItemButtonsBackgroundColor	
Description	Color of the background behind the item buttons
Syntax	HexColor
Default	EEEEEE
Context	Plug-in attribute

Color of the background behind the items buttons right of the bookmarks.

TextColor	
Description	Color of the hints (tool tips) displayed below the list
Syntax	HexColor
Default	000000
Context	Plug-in attribute

Color of the help texts below the bookmark list.

ScrollBaseColor, ScrollArrow, ScrollTrack	
Description	Colors for the scroll bar
Syntax	HexColor
Default	---
Context	Plug-in attribute

ScrollBaseColor defines the base color for the scroll bar in the bookmark list.
Additionally you specify a HexColor value for the Arrows and the track area of the scroll bar.

EnableAddFullBookmark, EnableAddLinkBookmark, EnableAddCustomBookmark, EnableSendBookmarks	
Description	Show or hide the buttons
Syntax	Boolean
Default	Depending on loaded add-on (SendBookmarks: false)
Context	Plug-in attribute

Use these parameters to enable or disable the buttons below the bookmark list.

- EnableAddFullBookmark: Enable bookmarking of entire images or pages
- EnableAddLinkBookmark: Enable bookmarking of links
- EnableAddCustomBookmark: Enable bookmarking of custom areas
- EnableSendBookmarks: Enable sending bookmark data

ImageEffects	
Description	Image modification parameters
Syntax	String
Default	---
Context	Plug-in attribute

Optional parameter to add image parameters for the thumbnail images. For FSI Server you can for example sharpen an image or adjust the image compression level, e.g. "effects=sharpen(230)&quality=95".

ExpandSelectionToImage	
Description	Expand image selections to fill entire thumbnail
Syntax	Boolean
Default	true
Context	Plug-in attribute

By default the notepad plug-in expands bookmark selections (custom and link bookmarks) to fill the entire thumbnail area. Setting the value to "false" disables expanding bookmark selections.

AddButton	
Description	Add notepad button to menu bar
Syntax	Boolean
Default	true (FSI Pages: false)
Context	Plug-in attribute

Add a button to the FSI Viewer menu bar to show or hide the notepad. FSI Pages skins provide a button independent of this parameter.

MenuOffset	
Description	Offset of the notepad button
Syntax	Number (pixels)
Default	0
Context	Plug-in attribute

Offset of the notepad button in the menu bar (FSI Viewer only).

Plug-in Parameters – Loading and Posting Data

The parameters below are required only if you want to send or load bookmark data to or from a web server.

Loading bookmarks from an XML file is as easy as specifying the URL to the XML data file using the parameter "LoadXML". You can use the "LoadXMLPrefix" parameter to prefix all "LoadXML" parameters, e.g. with the server URL.

Sending the XML data to a web server:

- Make sure to enable the "EnableSendBookmarks" parameter to display the corresponding button below the bookmark list
- If the user clicks the button, the plug-in sends the notepad data in XML format via HTTP POST to the server. The data is being send in the "data" post variable
- The answer (the content of the receiving page) of the web server in return must be simple XML data:

  ```
  <result success="true">filename.xml</result>
  ```

  if the server succeeded in saving the file OR

  ```
  <result success="false">error reason</result>
  ```

Depending on the result the plug-in displays the web page specified by the "SendXMLUrlSuccess" or "SendXMLUrlFailed".

The plug-in posts the following values to the success or failure HTML page:

Data	Description
datafile	The file name of the stored XML file returned by the server
error	Error description
FSI_Url	The URL of the calling FSI Viewer instance
FSI_notepad_id	The unique id of the notepad (UniqueID)

LoadXML	
Description	URL or file name of an XML data file to load
Syntax	String
Default	---
Context	Plug-in attribute

URL or file name of an XML data file to load from a web server.
If you add this parameter, the notepad loads the bookmarks from the file specified instead of loading bookmarks from the user's system.

LoadXMLPrefix	
Description	Prefix to use for the LoadXML parameter
Syntax	String
Default	---
Context	Plug-in attribute

You can use this value e.g. in the _default.fsi configuration file to specify the server address and folder to load XML notepad data files from.
Example: "http://foo.com/notepaddata/".

SendXMLStoreURL	
Description	URL to post notepad data to
Syntax	String
Default	---
Context	Plug-in attribute

URL of a web server receiving the notepad data in XML format.
See "Loading and Posting Data" on the preceding page for details.

SendXMLUrlSuccess	
Description	URL to open if sending XML data succeeded
Syntax	String
Default	---
Context	Plug-in attribute

Url to open if sending the XML data to the web server specified by "SendXMLStoreURL" parameter succeeded.

SendXMLUrlFailed	
Description	URL to open if sending XML data failed
Syntax	String
Default	---
Context	Plug-in attribute

URL to open if sending the XML data to the web server specified by "SendXMLStoreURL" parameter failed.

SendXMLUrlSuccessTarget, SendXMLUrlFailedTarget	
Description	HTML target frame to open the URL in
Syntax	String
Default	_blank
Context	Plug-in attribute

HTML target frame to open the "SendXMLUrlSuccess" or the "SendXMLUrlFailed" page in.

Plug-in Parameters – Advanced Customization_Customizing Image

You can replace the default animations / images displayed when the bookmark list is empty, when the user adds a custom or a link bookmark.

The plug-in loads the optional custom help images from your imaging server.
The syntax of these parameters therefore corresponds to the "FPXSrc" parameter.
If you specified the "FPXBase" parameter you only need to provide the path to the image, e.g. "helpimages/emptynotepad.tif".

The following images can be defined:
ImageHelpMain: image to display when the bookmark list is empty
ImageHelpLinkBookmark: image to display while the user adds a link bookmark in FSI Pages
ImageHelpLinkCustom: image to display while the user adds a custom bookmark

Plug-in PagesMirror

Plug-in Target:
Viewport of FSI Pages

Plug-in Location:
/plugins/pagesmirror.plg

Function:
Displays a mirror like effect.

Syntax:
```
<Plugin src="pagesmirror"/>
```

Plug-in Parameters

Blur	
Description	Defines the blur effect of pages
Syntax	Number
Default	---
Context	PagesMirror
Version	5.0.0

Defines the blur factor of the mirrored images.

Offset	
Description	Defines distance between the pages and the mrror effect
Syntax	Number (Pixels)
Default	---
Context	PagesMirror
Version	5.0.0

Defines the distance of the mirror effect and the pages.

StartAlpha	
Description	Alphablending
Syntax	Number
Default	70
Context	PagesMirror
Version	5.0.0

Defines the mirror effects opacity on the near end.

Plug-in PageSounds

Plug-in Target:
FSI Pages

Plug-in Location:
/plugins/pagesounds.plg

Function:
Adds page flip sounds to FSI Pages

Syntax:
```
<Plugin src="pagesounds" />
```

This plug-in adds page flip sounds to FSI Pages and does not have any effect with FSI Showcase or FSI Viewer instances.

Plug-in Parameters

Volume	
Description	Volume of the page flip sounds
Syntax	Number
Default	50
Context	Plug-in tag

The volume to play the flip sounds at from "0" (off) to "100" (maximum volume).

Library	
Description	File name of the sound library to use
Syntax	File Name
Default	pagesoundlib_coated.swf
Context	Plug-in tag

File name of an SWF file (compiled Flash movie) containing the flip sounds.
FSI Viewer ships with two different sound libraries:

pagesoundlib_coated.swf (coated pages like catalogs)
pagesoundlib_newspaper.swf (uncoated pages like newspapers)

Plug-in PagesThumbBar

Plug-in Target:
Viewport of FSI Pages

Plug-in Location:
/plugins/thumbbar.plg

Function:
Displays a bar with thumbnail images.

Syntax:
```
<Plugin src="thumbbar"/>
```

Plug-in Parameters

AlignLeft	
Description	Keep current image aligned to the left
Syntax	Number (Pixels)
Default	false
Context	PagesThumbBar
Version	5.0.0

When enabled, the thumbnail of the currently active image is shown left in the thumbnail bar. When disabled, the thumbnail is shown in the center below the catalog.

Arrows	
Description	Display scroll arrows
Syntax	Boolean
Default	true
Context	PagesThumbBar
Version	5.0.4

If set to false, the arrows for scrolling the thumbnailbar are hidden and the user needs to drag the thumbs to scroll

ArrowAlphaActive	
Description	Opacity of arrow
Syntax	Number
Default	100
Context	PagesThumbBar
Version	5.0.0

Opacity of the arrows when mouse hovers over the thumbnail bar.

ArrowAlphaNormal	
Description	Opacity of arrow
Syntax	Number
Default	0
Context	PagesThumbBar
Version	5.0.0

Defines the opacity of the arrows in normal state.

ArrowFillColor	
Description	Arrow fill color
Syntax	HexColor
Default	FFFFFF
Context	PagesThumbBar
Version	5.0.0

The fill color of the arrows inside in the thumbnailbar.

ArrowLineColor	
Description	Arrow line color
Syntax	HexColor
Default	FFFFFF
Context	PagesThumbBar
Version	5.0.0

The fill color of the arrows inside in the thumbnailbar

DropShadow	
Description	Enable Thumbnail Shadows
Syntax	Boolean
Default	false
Context	PagesThumbBar
Version	5.0.0

Enable/Disable the drop shadow effect of the thumbnails.

Height	
Description	Size of thumbnail images
Syntax	Number (Pixels)
Default	80
Context	PagesThumbBar
Version	5.0.0

Defines the height of the thumbnails in the thumbnail bar. The width is derived from the image size.

HideOnZoom	
Description	Hide thumbnails when zooming
Syntax	Boolean
Default	false
Context	PagesThumbBar
Version	5.0.0

When enabled, the thumbs will be hidden if an image is zoomed into.

MagnifyCurrentPages	
Description	Size of thumbnail images
Syntax	Float
Default	100
Context	PagesThumbBar
Version	5.0.0

Defines the magnification level of the current thumbnail. Values greater than 100 will increase, smaller values will decrase the thumbnail size.

MarginBottom	
Description	Margin of the thumb bar
Syntax	Number (Pixels)
Default	0
Context	PagesThumbBar
Version	5.0.0

Defines the bottom margin of the thumbnail bar.

MarginTop	
Description	Margin of the thumb bar
Syntax	Number (Pixels)
Default	8
Context	PagesThumbBar
Version	5.0.0

Defines the top margin of the thumbnail bar.

OffsetLeft	
Description	Offset of thumbnail bar
Syntax	Number (Pixels)
Default	0
Context	PagesThumbBar
Version	5.0.0

Defines the offset of the thumbnail bar.

OffsetRight	
Description	Offset of thumbnail bar
Syntax	Number (Pixels)
Default	0
Context	PagesThumbBar
Version	5.0.0

Defines the offset of the thumbnail bar.

Padding	
Description	Thumbnail Padding
Syntax	Number (Pixels)
Default	10
Context	PagesThumbBar
Version	5.0.0

Defines the distance between the thumbnails.

SelectionBorderWidth	
Description	Border of the active thumbnail
Syntax	Number
Default	4
Context	PagesThumbBar
Version	5.0.4

Width of the border around the active thumbnail.

SelectionColor	
Description	Color of the active thumbnail
Syntax	HexColor
Context	PagesThumbBar
Version	5.0.4

Defines the backgroundcolor of the active pages in the thumbnail bar.

SelectionMarginBottom	
Description	Bottom margin of the active thumbnail
Syntax	Number
Default	6
Context	PagesThumbBar
Version	5.0.4

Defines the bottom margin of the active thumbnail.

SelectionRoundCorner	
Description	Round corner size of active thumbnail
Syntax	Number
Default	4
Context	PagesThumbBar
Version	5.0.4

Defines the rounded corner effect for the active thumbnail. Straight corners if set to 0.

TextColor	
Description	Color of page numbers
Syntax	HexColor
Context	PagesThumbBar
Version	5.0.4

Defines the rounded corner effect for the active thumbnail. Straight corners if set to 0.Defines the color of the page numbers in the Thumbbar Plugin.

TextColorHighlight	
Description	Round corner size of active thumbnail
Syntax	Number
Context	PagesThumbBar
Version	5.0.4

Defines the textcolor of the active thumbnail.

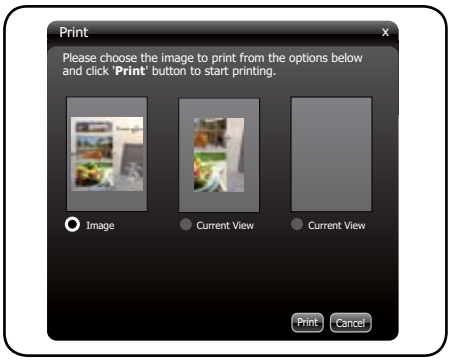
Plug-in PrintSave

Plug-in Target:
User interface

Plug-in Location:
/plugins/printsave.plg

Function:
Add printing and saving capabilities to FSI Viewer.

Syntax:
```
<Plugin src="printsave"/>
```



Plug-in Parameters

MenuOffset	
Description	Offset in pixel to previous menu elements
Syntax	Number (Pixels)
Default	---
Context	PrintSave
Version	5.0.0

Offset in pixel to previous menu elements.

Print	
Description	Enable/Disable printing
Syntax	Boolean
Default	---
Context	PrintSave
Version	5.0.0

If disabled, the menu button is hidden.

PrintEffects	
Description	Image effects
Syntax	String (urlencoded)
Default	---
Context	PrintSave
Version	5.0.0

Defines image modification parameters to apply to images used for printing. The value of this parameter depends on the imaging server being used.
With FSI Server you can sharpen the image and define the JPEG compression level, e.g. "effects=sharpen(230)&quality=95".

PrintResolution	
Description	Max. size of image to print
Syntax	Number (Pixels)
Default	2000
Context	PrintSave
Version	5.0.0

Defines the maximum size of the image in pixels to download for printing.
Please note that the actual size depends on the size of the image and (with FSI server) on the maximum size defined in the real time template used with FSI Viewer (see parameter PrintTemplate).

PrintTemplate	
Description	Image template to be used
Syntax	String
Default	---
Context	PrintSave
Version	5.0.0

Defines the FSI Server profile or eRez real time template used to load the print image.

Save	
Description	Enable/Disable saving
Syntax	Boolean
Default	true
Context	PrintSave
Version	5.0.0

If disabled, the menu button is hidden.

SaveEffects	
Description	Image effects
Syntax	String (urlencoded)
Default	---
Context	PrintSave
Version	5.0.0

Defines image modification parameters to apply to images used for saving. The value of this parameter depends on the imaging server being used.
With FSI server you can sharpen the image and define the JPEG compression level, e.g. "effects=sharpen(230)&quality=95".

SaveResolution	
Description	Max. size of image to save
Syntax	Number (Pixels)
Default	600
Context	PrintSave
Version	5.0.0

Defines the maximum size of the image in pixels to download for printing.
Please note that the actual size depends on the size of the image and (with FSI server) on the maximum size defined in the real time template used with FSI Viewer (see parameter PrintTemplate).

SaveTemplate	
Description	Image template to be used
Syntax	String
Default	---
Context	PrintSave
Version	5.0.0

Defines the FSI Server profile or eRez real time template used to download an image.

Plug-in Resize

Plug-in Target:
User interface

Plug-in Location:
/plugins/resize.plg

Function:
Rearranges user interface elements after resizing the viewer

Syntax:
```
<Plugin src="resize" MaxWidth="800 MaxHeight="600"/>
```

Using the resize plug-in you can define the viewer dimension in percent of your HTML document. After resizing the plug-in rearranges the user interface according to the size of the object tag without reloading the entire viewer instance and without the need for client side scripting. Note that this plug-in is required only, if you want to allow the user to dynamically resize an FSI Viewer instance.

Plug-in Parameters

MaxWidth and MaxHeight	
Description	Maximum width and height
Syntax	Number in pixel
Default	—
Context	Plug-in Tag

FSI Viewer can not be resized to a dimension greater than specified by these values.

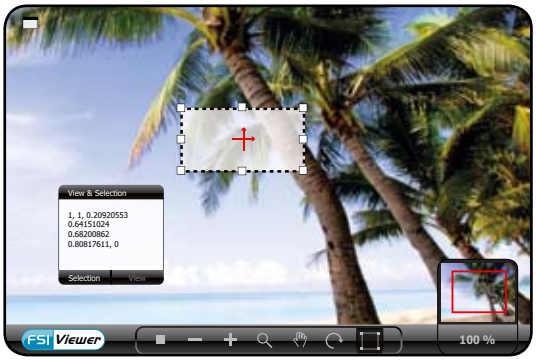
MinWidth and MinHeight	
Description	Minimum width and height
Syntax	Number in pixel
Default	—
Context	Plug-in Tag

FSI Viewer can not be resized to a dimension smaller than specified by these values.

EnlargeBy	
Description	Resize viewer in steps of *n* pixels
Syntax	Number
Default	1
Context	Plug-in tag

Specifies the steps FSI Viewer will be resized by. Entering 1 means that the viewer resizes exactly according to the size of the object tag. Entering greater numbers forces the viewer to resize by the number of pixels specified. Using values > 1 increases the chance of imaging server cache hits and might thus reduce server load.

Plug-in SelectFrame



Plug-in Target:
User interface

Plug-in Location:
/plugins/selectframe.plg

Function:
Provides a resizable select frame

Syntax:
`<Plugin src="selectframe" />`

This plug-in adds a resizable select frame into FSI Viewer.

Pressing SHIFT locks the aspect ratio of the selection.
Pressing CTRL resizes the selection relative to the center
Pressing ALT rotates the selection
Pressing ALT + SHIFT rotates the selection in steps of 5 degrees

Plug-in Parameters

The following attributes can be assigned to the <plugin> tag.
You might as well define all plug-in parameters via query or inside the <options> group by using the prefix "`selectframe_`" with the parameter name, e.g. "`selectframe_window`".

MenuOffset	
Description	Indentation of the menu button
Syntax	Number
Default	0
Context	Plug-in attribute

Specifies the space in pixels left of the menu button of the plug-in.

Visible	
Description	Show or hide the plug-in on startup
Syntax	Boolean
Default	false
Context	Plug-in attribute

Show or hide the select box on startup.

Window	
Description	Show or hide the output window
Syntax	Boolean
Default	true
Context	Plug-in attribute

Show or hide the output window of the plug-in.

AnimatedRect	
Description	Enable or Disable frame animation
Syntax	Boolean
Default	false
Context	Plug-in attribute

If enabled the frameborders will be animated.

Rotate	
Description	Enable rotation of the selection
Syntax	Boolean
Default	true
Context	Plug-in attribute

Enable rotating the selection by pressing ALT key or ALT + SHIFT keys and dragging the handles of the select box.

MaskColor	
Description	Color of the mask
Syntax	HexColor
Default	FFFFFF
Context	Plug-in attribute

Define a 6-digit hexadecimal color value for the mask of the select box.

MaskAlpha	
Description	Opacity of the mask
Syntax	Number
Default	75
Context	Plug-in attribute

Defines the opacity of the mask of the select box (0 = invisible).

AspectRatio	
Description	Aspect ratio of the selection
Syntax	String
Default	---
Context	Plug-in attribute

Defines an aspect ratio for the selection, e.g. "320:200".
Use SHIFT to lock or unlock the aspect ratio when resizing the selection.

InvertShift	
Description	Inverts the function of the SHIFT key
Syntax	Boolean
Default	false
Context	Plug-in attribute

You can use this parameter to invert the function of the SHIFT key while dragging.

Callback	
Description	JavaScript callback function
Syntax	String
Default	---
Context	Plug-in attribute

Specify the name of a custom JavaScript function to call each time the selection changes. The arguments passed to this function are:

SceneSet, Scene, Left, Top, Right, Bottom, Rotation, Current View

Example: JavaScript callback function

```
function onSelect(set, scene, l, t, r, b, rot, strView){
       doSomething();
}
```

Additionally you can as use the template "$$`selection`$$" for the *CustomButton* plug-in.

Events	
Description	Send plug-in event notifications
Syntax	Boolean
Default	false
Context	Plug-in attribute
Version	3.2.5 and above

Specifies whether the plug-in send plug-in event notifications. Plug-in events can be handled when using JSBridge plug-in or FSI Component.

The following events apply to this plug-in:

Event ID	Parameter	Description
onSelectFrameEnable	Boolean	The select has been enabled or disabled.
onSelectFrameStart	---	The user starts selecting.
onSelectFrame	String	The user modifies the select frame. The parameter provides a string representing the current selection.
onSelectFrameStop	String	The user stopped modifying the selection. The parameter provides a string representing the current selection.

Plug-in ShoppingList

Plug-in Target:
User interface of FSI Pages

Plug-in Location:
/plugins/shoppinglist.plg

Function:
Add shopping cart functionality to FSI Pages.

Syntax:
```
<Plugin src="shoppinglist"/>
```

Plug-in Parameters

Currency	
Description	Defines the currency
Default	---
Context	ShoppingList
Version	5.0.0

Defines the currency inside the shoppinglist.

DecimalCharacter	
Description	Decimal delimeter
Syntax	String
Default	---
Context	ShoppingList
Version	5.0.0

Decimal delimeter.

DefaultUnit	
Description	Default unit string
Syntax	String
Default	pce
Context	ShoppingList
Version	5.0.0

Defines the default unit of items added to the shoppinglist.

Email	
Description	Enable/Disable sending list by email
Default	true
Context	ShoppingList
Version	5.0.0

When enabled, the user can email a shoppinglist to others.

MaxCountPerItem	
Description	Maximum Count per item
Syntax	Number
Default	99
Context	ShoppingList
Version	5.0.0

Defines the maximum unit per item that can be added to the shoppinglist.

MinItemHeight	
Description	Minimum Height of items in the list
Syntax	Number
Default	64
Context	ShoppingList
Version	5.0.0

Defines the minimum height of each item in the list.

PersistentSorage	
Description	Minimum Height of items in the list
Default	true
Context	ShoppingList
Version	5.0.0

Enable or disable the caching of list data. If disabled, the list info is lost when the user leaves the FSI pages instance.

PersistentStorageExpiresAfter	
Description	Expire list data
Syntax	Number
Context	ShoppingList
Version	5.0.0

Defines after which period of time the shopping list data expires.

Print	
Description	Minimum Height of items in the list
Default	true
Context	ShoppingList
Version	5.0.0

Enable/Disable printing.

ShowListOnAddItem	
Description	Show shoppinglist behaviour
Syntax	string
Default	once
Context	ShoppingList
Version	5.0.0

Defines wether the shoppinglist is displayed after an item has been added.

TargetURL	
Description	Defines the target for shoppinglist
Default	---
Context	ShoppingList
Version	5.0.0

Defines the target URL of the servlet the shoppinglist data is send to.

Template	
Description	Template to use
Syntax	template name
Context	ShoppingList
Version	5.0.0

Defines the HTML templates to be used for rendering the pages.

ShowListOnAddItem	
Description	Show shoppinglist behaviour
Default	once
Context	ShoppingList
Version	5.0.0

Description of the template.

DialogHeight	
Description	Defines the height of the shoppinglist dialog window
Syntax	Number
Default	400
Context	ShoppingList
Version	5.0.0

Defines the height of the shoppinglist dialog window.

DialogWidth	
Description	Defines the width of the shoppinglist dialog window
Syntax	Number
Default	600
Context	ShoppingList
Version	5.0.0

Defines the width of the shoppinglist dialog window.

Plug-in SoftwareCursor

Plug-in Target:
User interface

Plug-in Location:
/plugins/softwarecursor.plg

Function:
Adds a software cursor to the FSI Viewer.

Syntax:
```
<Plugin src="softwarecursor" />
```

The "Software Cursor" plug-in adds a software cursor representing the current mouse mode (Zoom, Drag, Rotate-2D, Rotate 3D).

Plug-in parameters

Library	
Description	File name of an SWF containing cursor graphics
Syntax	String
Default	— (built-in library)
Context	Plug-in attributes

File name of an SWF (compiled Flash movie) file containing custom cursor graphics. Please refer to the *FSI Contrib package* for a sample and instructions how to create your own cursor library.

Plug-in StickyNotes

Plug-in Target:
User interface of FSI Pages

Plug-in Location:
/plugins/stickynotes.plg

Function:
Adds bookmark functionality to FSI Pages.

Syntax:
```
<Plugin src="stickynotes"/>
```

Plug-in Parameters

AllowMirroredNotes	
Description	Enable/Disable mirrored note text
Syntax	Boolean
Default	false
Context	StickyNotes
Version	5.0.0

When enabled, the notes text will be mirrored when the page is turned, otherwise it will not be displayed.

Alpha	
Description	Opacity of the sticky notes
Syntax	Number
Default	50
Context	StickyNotes
Version	5.0.0

Defines the opacity of the sticky notes from invisible (0) to opaque (100).

Color	
Description	Defines the color of the sticky notes
Syntax	Color
Default	FFFF55
Context	StickyNotes
Version	5.0.0

Defines the main color of the sticky notes.

ColorHighlight	
Description	Color of lighter part
Syntax	Color
Default	FFFFEE
Context	StickyNotes
Version	5.0.0

Defines the color of the light part of the sticky notes.

ColorShadow	
Description	Defines the color of the dark part of the sticky notes
Syntax	Color
Default	CCCC00
Context	StickyNotes
Version	5.0.0

Defines the color of the dark part of the sticky notes.

NaturalRotation	
Description	Enables slightly rotated sticky notes
Syntax	Boolean
Default	true
Context	StickyNotes
Version	5.0.0

When this option is enabled, the sticky notes will be rotated slightly, to give a more natural look and feel.

PageNumbers	
Description	Enable/Disable Page numbers
Default	True
Context	StickyNotes
Version	5.0.0

Display page numbers on the sticky notes.

PersistentStorage	
Description	Cache notes inforation
Default	True
Context	StickyNotes
Version	5.0.0

When disabled, the sticky notes will be deleted when the user leaves the FSI Pages instance. When enabled the data will be cached for the given time (defined by PersistentStorageExpiresAfter).

PersistentStorageExpiresAfter	
Description	Keep note data for given time
Syntax	Number
Context	StickyNotes
Version	5.0.0

Defines after which period of time the cached note data expires.

Position	
Description	Position of the sticky notes
Syntax	String
Default	top
Context	StickyNotes
Version	5.0.0

Sticky notes position on the page (top, bottom or right).

Scale	
Description	Size of the sticky notes
Syntax	Number
Default	100
Context	StickyNotes
Version	5.0.0

Defines the size of the sticky notes.

Spacing	
Description	Defines the distance between the sticky notes
Syntax	Number (Pixels)
Default	8
Context	StickyNotes
Version	5.0.0

Defines the distance between the sticky notes.

Plug-in Synchronize

Plug-in Target:
N/A

Plug-in Location:
/plugins/synchronize.plg

Function:
Synchronize multiple instances of FSI Viewer.

Syntax:

```
<Plugin src="synchronize" ID="UNIQUE_STRING" />
```

Enterprise editions of FSI Viewer contain this plug-in by default.
For all other editions this plug-in can be optionally obtained.

Using the Synchronize plug-in you can synchronize multiple instances of FSI Viewer. To synchronize FSI Viewers you need to use this plug-in in all instances of FSI Viewer you want to synchronize and assign the same unique ID parameter to the instances.
The plug-in synchronizes the view (2D and 3D) as well as the mouse cursor and the plug-in measure. The plug-in does not synchronize add-ons like FSI Showcase or FSIPages.

Plug-in Parameters

ID	
Description	Unique identifier
Syntax	String
Default	---
Context	Plug-in attribute

All FSI Viewer instances using this plug-in with the same ID will be synchronized. Please note that this will even be the case if the viewers reside in different browser windows.

SyncSoftwareCursor	
Description	Synchronize SoftwareCursor plug-in
Syntax	Boolean
Default	true
Context	Plug-in attribute
Version	4.1.5 or above

Enable/disable synchronizing the mouse cursor position among synchronized FSI Viewer instances. Note that this parameter has no effect, if you do not use the SoftwareCursor plug-in.

SyncMeasure	
Description	Synchronize the Measure plug-in
Syntax	Boolean
Default	true
Context	Plug-in attribute
Version	4.1.5 or above

Enable/disable synchronizing the Measure plug-in. This parameter has no effect, if you do not use the Measure plug-in. You might want to disable this parameter if you want to enable users to measure distances in synchronized FSI Viewer instances independently.

Plug-in TextBox

Plug-in Target:
User interface

Plug-in Location:
/plugins/textbox.plg

Function:
Adds a text box to display text related with the image below the viewer.

Syntax:
```
<Plugin src="textbox"/>
```

This plug-in adds a text box below the viewer to display text related with the image. With FSI Server or eRez Imaging Server you can directly access IPTC data of the image to retrieve the text automatically without the need for a database. When using FSI Showcase the text updates each time you select another image.

The following HTML tags can be used to format the text:

```
<b>…</u> <i>…</i> <u>…</u>
<br/>
<font size="n" color="#RRGGBB" >…</font>
<p align="…">…</p>
<a href="…"/>…</a>
```

Plug-in Parameters

TextFrom	
Description	Parameter containing the text data
Syntax	String
Default	IPTC_Caption
Context	Plug-in attribute

Defines the name of the FSI Viewer parameter (in the <options> section of FSI Viewer configuration) containing the text to display. You can specify short text as an attribute of this parameter, e.g. <IPTC_Data value="Some Text" />.
For multi-line text or text containing HTML tags you can alternatively use a text node: <IPTC_Data>Text line 1.
Text line 2.</IPTC_Data>

TextColor	
Description	Default text color
Syntax	HexColor
Default	000000
Context	Plug-in attribute

Default text color. To use different text colors you can use the HTML tag <font color="RRGGBB"/>.

TextSize	
Description	Default text size
Syntax	Number
Default	11
Context	Plug-in attribute

Default text size. To use different sizes in the text you can use the HTML tag <font size="n"/>.

Height	
Description	Height of the text box
Syntax	Pixels
Default	60
Context	Plug-in attribute

Height of the text box in pixels.

BGColor	
Description	Background color of the text box
Syntax	HexColor
Default	FFFFFF
Context	Plug-in attribute

Background color of the text box.

BGAlpha	
Description	Opacity of the text box
Syntax	Number
Default	100
Context	Plug-in attribute

Opacity of the background from 0 (invisible) to 100 (opaque).

BaseColor	
Description	Base color for the interface parts
Syntax	HexColor
Default	DDDDDD
Context	Plug-in attribute

Base color to use for the interface parts of the plug-in, e.g. the scroll bar and the border.

Border	
Description	Display a border to separate the text from the view port
Syntax	Boolean
Default	true
Context	Plug-in attribute

Display a border separating the text from the viewer's view port.

BorderColor	
Description	Color of the border
Syntax	HexColor
Default	---
Context	Plug-in attribute

Color of the border separating the text from the viewer's view port.

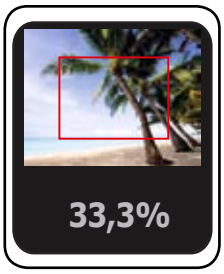
BorderWidth	
Description	Width of the border
Syntax	Number
Default	2
Context	Plug-in attribute

Width of the border separating the text from the view port.

ScrollArrowColor	
Description	Color of the arrows of the scroll bar
Syntax	HexColor
Default	Derived from BaseColor parameter
Context	Plug-in attribute

6-digit hexadecimal color value for the arrows in the scroll bar.

Plug-in ZoomMeter



Plug-in Target: User interface

Plug-in Location:
/plugins/zoommeter.plg

Function:
Displays the current level of magnification.

Syntax:
```
<Plugin src="zoommeter" color="000000" />
```

The "ZoomMeter" plug-in extends the navigator window with a display of the current magnification.

Plug-in Parameters

Color	
Description	Text color
Syntax	HexColor
Default	"000000" (black)
Context	Plug-in attribute

Text color of the zoom display.
The color has to be defined as a 6-digit hexadecimal number.
(E.g. "FFFF00" for yellow).

SrcRelative	
Description	Display magnification relative to source image or viewer size
Syntax	Boolean
Default	false
Context	Plug-in attribute
Version	3.0.0 or higher

Display the current magnification in percent of the viewer size (false) or relative to the size of the source image.

Error Messages

Please make sure to enable the *debug mode* when experiencing problems or to check FSI Viewer configuration!

Usually it is easy to locate the reason for an error by examining the output in the debug window or the step of the initialization process where the FSI Viewer stops.

Warning: ViewerWidth and ViewerHeight obsolete

FSI Viewer version 4 determines the dimension of the viewer instance automatically. Therefore specifying these values is not required any more. Specifying the correct viewer dimension does on the other hand not have any negative influence.

Error: FSI Viewer does not show up at all

Please make sure to have Flash plug-in 6,0,65,0 or above installed.

Please check the parameters of the <object> and <embed> tag.
Make sure that the HTML code:

```
<PARAM NAME="movie" VALUE="[URL and parameter]">
```

contains the correct URL to the FSI Viewer file "fsi.swf".

Please make sure that all required files have been transferred in binary mode to your web server.
If not, adjust the corresponding configuration options of your FTP client accordingly and upload all FSI Viewer files once again.

Error: Loading _default.fsi…failed

Please make sure that the file "_default.fsi" resides in the setup directory of your FSI Viewer.
Check the _default.fsi file for XML syntax errors.

Error: Loading *.fsi…failed
or
Error: FPXSrc undefined

Please make sure that the configuration file defined by the *CFG parameter* is valid.
Please make sure that the *FSIBase parameter* in the _default.fsi file points to the correct directory.
Please check the XML syntax of the corresponding *FSI-file*.

Error: The Viewer stops displaying the message "Loading Skin and FSI Module:"

Please make sure that all files have been uploaded to the location provided when ordering your FSI product. Check the "skin" parameter and make sure it contains an existing skin file name without any file extension.

Error: The Viewer stops displaying the message "Opening FPX: […]"

Please check the given path to the source image file on the imaging server.
Adjust the parameters FPXSrc or FPXBase accordingly.

Error: Viewer stops displaying the message "Loading Plug-in …"

Please check the "Src" parameter of the specified plug-in.
Make sure that the corresponding plug-in file exists in the "/plugins" subdirectory of your setup directory of your web server.
Error: The image appears entirely black or distorted

This might happen due to an error during the conversion of your image to the FPX format.
Please make sure that the source image has not been saved in grayscale mode and that it does not contain multiple alpha channels.
Be sure to save the image without FPX data compression.
Please try converting your image to RGB color mode and remove all alpha channels prior to FPX conversion.

Error: The Viewer displays multiple images at a time

You might be using an FPX containing multiple images without defining the parameters TilesX and TilesY or these parameters have been defined incorrectly.
Please check the output of the *Debug Window* for the *SceneSets* setting.

Error: Keyboard commands not working

Using the keyboard commands require the FSI Viewer to have the input focus.
Click anywhere inside the Viewer to pass the focus to the FSI Viewer.

Appendix

FSI Contrib Package

The FSI Contrib package contains sample configuration files and Adobe Flash files you can use to customize FSI Viewer. Please contact your FSI Viewer dealer if you did not receive the FSI Viewer Contrib package along with your copy of FSI Viewer.

Escaping or URL-encoding parameter values

In some situations you need to pass escaped (URL-encoded) parameters values to FSI Viewer, because using unescaped values would break the XML syntax or because you need to use characters prohibited in URLs.
In cases like this you can pass escaped values rather than passing the actual value.

What does escaping (URL-encoding) mean?
URL-encoding (or percent-encoding) means converting the character to its corresponding value in ASCII and then representing that value as a pair of hexadecimal digits. The digits, preceded by a percent sign ("%"), are then used in place of the reserved character. FSI Viewer transforms URL-encoded values back to the original value.

How can I URL-encode a value?
Most scripting languages (JavaScript, ActionScript, PHP and alike) provide methods for this purpose. Please note that FSI Viewer from version 4 requires Unicode safe encoding rather than non-unicode safe encoding – this is an escaped character meight be represented by one or more 2-digit numbers preceded by "%".

Using JavaScript you therefore need to use the method "encodeURIComponent()" instead of "escape()".

If you don't have a tool handy to URL-encode a value, you might want to use the tiny JavaScript based tool "encoding/urlencode.html" contained in the FSI Contrib package.

HTML Tags available in FSI Viewer

FSI Viewer from version 4 supports basic HTML tags in various contexts like when entering custom tool tips, texts for the "LargeToolTips" plug-in, search results and alike. Whenever this documentation refers to "Basic HTML formatting tags", the following HTML tags can be used (variable values printed bold italic):

```
<b></b>
<i></i>
<u></u>
<br/>
<font color="#FF0000" size="12" face="Arial"></font>
<p align="left|center|right"></p>
```

Please note that you need to use XML syntax in order to avoid XML syntax errors in configuration files - this is e.g. using <br/> instead of <br>.

Example of a _default.fsi file

```
<fsi_parameter>
	<!-- This file contains default parameters.
	All parameters will be overwritten by additional .fsi files or query
parameters. -->

	<FPX>
		<ServerType value="fsi" />
		<Base value="http://fsi.neptunelabs.com/fsi/server?source=/" />
	</FPX>

	<Plugins>
		<Plugins src="mousemodes" />
		<PLUGIN src="zoommeter" SrcRelative="1"/>
	</Plugins>

	<Options>
		<FSIBase value="config/" />
		<ScenePreload value="true" />
		<MenuAlign value="BR" />
	</Options>
</fsi_parameter>
```

Example of an image specific FSI configuration file

```
<fsi_parameter>

	<FPX>
		<!-- Src will be prefixed by FPXBase of _default.fsi -->
		<Src		value="3d_object.fpx" />
		<Width			value="8128" />
		<Height			value="9168" />
		<TilesX			value="4" />
		<TilesY			value="3" />
	</FPX>

	<Options>
		<InitialAction value="NextScene" />
		<InitialActionDelay value="3" />
		<MenuAlign value="TL" />
		<NoNav value="true" />
	</Options>

</fsi_parameter>
```

Credits

Special thanks go to the following persons for their valuable support such as providing translations, sharing their ideas regarding improvements and beta testing:

Bora Alioglu
Oscar Applegren
Sven-Erik Andersen
Márton Balázs
Radu Basca
Radoslav Bielik
Simone Chiaretta
Maciej Chojnacki
Jacky Choy
Petr Čihař
Jesús M. Delgado
Marc Delisle
Daniel Hinostroza
Linas Jankauskas
Yukihiro Kawada
Slawomir Kilanowski
Hillar Mäeotsa
Kristian Ottosen
Ivo Teel
Stas Ukolov
Andri Viiand

Index

FSI Viewer – FSI Pages - FSI Showcase
Flash based Single Source Image Viewer

NeptuneLabs GmbH

Detmolder Str. 210a
32791 Lage
Germany

Fon: +49 (0) 5232-9999 7-0
Fax: +49 (0) 5232-9999 7-29
eMail: info@neptunelabs.com
WWW: www.neptunelabs.com
www.fsi-viewer.com

www.fsi-viewer.com

Neptune Labs
Internet Application Development

NeptuneLabs GmbH
Detmolder Str. 210a
32791 Lage
Germany
fon: + 49 5232 - 99997- 0
fax: + 49 5232 - 99997- 29
info@neptunelabs.com
www.neptunelabs.com